COMPUTER SCIENCE
TECHNICAL REPORT SERIES

UNIVERSITY OF MARYLAND
COLLEGE PARK, MARYLAND
20742

ALGORITHMS AND HARDWARE TECHNOLOGY
FOR IMAGE RECOGNITION

Quarterly Report
1 May-31 July, 1976

Contract DAAG53-76C-0138
(DARPA Order 3206)

Computer Science Center
University of Maryland
College Park, MD   20742

ABSTRACT

Techniques for detecting tactical targets on Forward-
Looking Infrared (FLIR) imagery are being investigated.
The principal topics covered include target and background
models, object extraction and classification, and hardware
technology applicable to real-time implementation.

DDC

RECEIVED

JAN 31 1977

A

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF TABLES

1. Introduction

This document reports on the progress of the University of Maryland/Westinghouse Corporation project entitled "Algorithms and Hardware Technology for Image Recognition" during the initial period May 1-July 31, 1976. The project has two principal goals:

    a) Selection of state-of-the-art algorithms for automatic target cueing, and implementation of one or two selected algorithms in hardware to demonstrate the feasibility of incorporating such algorithms in a reconnaissance sensor.

    b) Exploration of new approaches to image understanding, with emphasis on techniques applicable to target cueing and similar applications, as well as on image modeling for performance prediction.

The project consists of three phases all of which involve collaboration between the University and its subcontractor, the Systems Development Division of Westinghouse. The three phases and their breakdown into tasks are displayed in the following table:

Phase	Task
I	(Task and technology review)
	1) Data base acquisition
	Obtain data bases consisting of real-world imagery containing representative target-background combinations for selected reconnaissance sensors and conditions.
	2) Review of tri-service operational needs and resulting system design constraints.
	Meet with tri-service representatives to dis-

Phase	Task

cuss operational target detection problems. Emphasis will probably be placed on night vision and tactical IR sensors. Choice of sensors and operational environments will define constraints on hardware design.

3) **Hardware/algorithm interface**

Hardware constraints will restrict choice of algorithms for implementation; algorithm design will define requirements on hardware performance. This interaction will constitute a continuing aspect of the Maryland/Westinghouse collaboration.

II (Algorithm development and testing)

4) **Algorithm development**

Exploration of new approaches; evaluation of standard approaches, modified as appropriate for the given input data.

5) **Algorithm selection and test**

Algorithm implementation, feasibility testing, performance evaluation on selected data bases, comparison with current target cueing systems performance.

6) **Target and background modelling**

Development of statistical models; estimation of model parameters for given data bases; use of models to predict target detection performance.
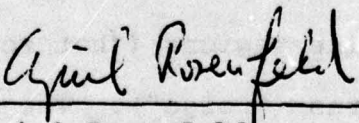
III (Hardware design, fabrication, and testing)
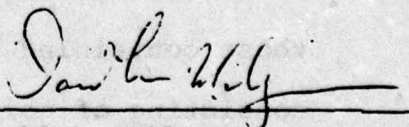
In the first quarter, major efforts have been devoted to a cross-sectional study of the algorithmic steps comprising a solution to the cueing problem. The purpose has been to investigate the inherent complexity of Forward-Looking Infrared (FLIR) imagery and to identify the areas in which significant contributions to the state of the art are likely to be made.

The current research effort in automatic target cueing consists of seven project areas:

. Data base acquisition and preprocessing

. Models for FLIR image understanding

. Automatic object detection

. Automatic threshold selection

. Noise region elimination and component feature extraction

. Component classification and target recognition

. Hardware technology for algorithm implementation

In each project area, one or more approaches have been studied as described in the following sections of this report. The Westinghouse review of hardware technology is appended to this report as a separate volume.

<u>Azriel Rosenfeld</u>
Azriel Rosenfeld
Research Professor
(Principal Investigator)

<u>David L. Milgram</u>
David L. Milgram
Assistant Professor
(Co-Principal Investigator)

2. Project Review

A1. Data Base Acquisition and Preprocessing

The image data base which has been investigated in this report consists of low altitude infrared scenes of tanks, trucks and APC's against a sparsely wooded or barren background. The images were digitized by the U.S. Army Night Vision Laboratory (NVL) from video tapes of the FLIR signal which drives the cockpit display. Westinghouse reformatted the data and supplied duplicate digital tapes. Aside from a variety of noise effects, the image display fiducial marks and numeric situation data. A number of scenes were imaged in complement (negative) format.

In all, 13 tapes containing 90 scenes were received. Each scene image was present as a tape file of 800 records (lines) of 1024 bytes (pixels) each. The pixels had been quantized to 16 bits. According to the ground truth supplied, the scenes contained views of targets in various aspects and at various ranges. The available ground truth is presented in Table 1.

Using the ground truth, a set of 128x128 pixel windows containing the identified targets were examined. In addition, a number of windows containing no targets were extracted. Among the latter, a distinction was made between those containing object-like regions ("hot rocks") and those consisting of noise patterns ("noise"). The windows were further reduced by sampling to 64x64 image points. The extracted windows were requantized to 64 gray levels by dropping the low order two bits. As may be seen from the typical

Image Ref.No.	Tape	Frame	Contents			
1	A	1	T/S			
2	A	2	T/S			
3	A	3	T/S	R/S		
4	A	4	T/S	R/S		
5	A	5	*			
6	A	6	T/S	R/S		
7	A	7	*			
8	A	8	T/S			
9	A	9	T/S	R/S		
10	A	10	T/S			
11	B	1	T/S			
12	B	2	T/S			
13	B	3	T/S			
14	B	4	T/S			
15	B	5	T/S			
16	B	6	T/S			
17	B	7	T/S			
18	B	8	R/E			
19	B	9	*			
20	B	10	*			
21	C	1	A/S	T/E		
22	C	2	A/S	T/E	R/E	
23	C	3	A/S	T/E	R/E	
24	C	4	A/S	T/E	R/E	
25	C	5	*			
26	C	6	R/E	T/S		
27	C	7	R			
28	C	8	R	T/S		
29	C	9	T/S	A		
30	C	10	T/S	A		
31	D	1	R	T/S		
32	D	2	R	T	A	
33	D	3	R	T	A	
34	D	4	R	T	A	
35	D	5	R	T	A	
36	D	6	*			
37	D	7	A/E			
38	D	8	A/E	T/S		
39	D	9	*			
40	D	10	T			
41	E	1	R/E			
42	E	2	A/E	T/D		
43	E	3	T/D			
44	E	4	A/E			
45	E	5	A/E	T/D		
46	E	6	A/E	T/D		
47	E	7	R/E			
48	E	8	A/E	R/D		
49	E	9	*			
50	E	10	T	A		
51	F	1	A	T/S	R	
52	F	2	A	T/S	R	
53	F	3	A	T/S	R	
54	F	4	A	T/S	R	
55	F	5	A	T/S	R	?
56	F	6	R	T	A	?
57	F	7	R	T	A	
58	F	8	R	T	A	?
59	F	9	R	T	A	?
60	F	10	A/E			
61	GH	1	A/E	T/D		
62	GH	2	T/S			
63	GH	3	T/S			
64	GH	4	T/S			
65	GH	5	?	T		
66	GH	6	?	T	?	
67	GH	7	?	?	?	
68	GH	8	?	T		
69	GH	9	?	T		
70	GH	10	*			
71	GH	11	R/D			
72	GH	12	R/D			
73	GH	13	A/E	T/D		
74	GH	14	A/E	T/D		
75	GH	15	A/E	T/D		
76	HI	1	A/E	T/D		
77	HI	2	R/D			
78	HI	3	A/E	T/D		
79	HI	4	A/E	T/D		
80	HI	5	A/E	T/D		
81	HI	6	?	T		
82	HI	7	?	T		
83	HI	8	A/E			
84	HI	9	A/E			
85	HI	10	A/E			
86	HI	11	A/E			
87	HI	12	A/E			
88	HI	13	A/E	T/D		
89	HI	14	T/D			
90	HI	15	A/E			

```
T = tank      /S = side view
R = truck     /E = end view
A = apc       /D = 3/4 view
* = no target
```

Table 1. NVL Data Ground Truth.

Image Ref.No.	Tape	Frame	Contents	
91	JK	1	A/E	
92	JK	2	T/D	
93	JK	3	A/E	
94	JK	4	A/E	
95	JK	5	T/D	
96	JK	6	A/E	
97	JK	7	A/E	
98	JK	8	A/E	
99	JK	9	T/D	
100	JK	10	R/E	
101	JK	11	A/E	
102	JK	12	A/E	
103	JK	13	A	
104	JK	14	R	
105	JK	15	T/D	
106	L	1	A/E	
107	L	2	A/E	T
108	L	3	A/E	T
109	L	4	T	R
110	L	5	T/D	
111	L	6	A/E	
112	L	7	A/E	
113	L	8	A/E	
114	L	9	T/D	A/E
115	L	10	A/E	
116	MN	1	A	
117	MN	2	A	
118	MN	3	A	
119	MN	4	A	
120	MN	5	A	
121	MN	6	A	
122	MN	7	T	A
123	MN	8	T	A
124	MN	9	T	
125	MN	10	T	A
126	MN	11	T	
127	MN	12	T	A
128	MN	13	T	
129	MN	14	T	A
130	MN	15	T	A
131	NO	1	R	
132	NO	2	T/E	
133	NO	3	R	T
134	NO	4	R	
135	NO	5	T/E	
136	NO	6	T/E	
137	NO	7	R	
138	NO	8	R	
139	NO	9	R	
140	NO	10	R	
141	NO	11	R	
142	NO	12	R	
143	NO	13	T	
144	NO	14	R	
145	NO	15	T	

Table 1 (continued)

histograms shown in Figure 1, the original images exhibit
non-uniformity of quantization, and the information loss
due to requantization should be small. A final preprocess-
ing step complemented those windows which contained targets
in complement form.

An assessment of the data base reveals a wide
range of target sizes and levels of thermal emission. To
the naked eye, some of the small indistinct targets, while
detectable, appear virtually unclassifiable. The larger
targets do exhibit characteristic shapes, though. We have
assumed at this stage that it is more important to detect
targets at long range than to classify them once their
shapes are discernible at closer range. However, shape
recognition for target classification will be studied ex-
tensively in the near future.

The variability of the images and the large
amount of noise present indicate the need for the acquisition
of further data bases to substantiate or challenge the
assumptions made in the present study.

A2. Image Processing Software

Software development has progressed in the imple-
mentation of MINIXAP, a research-oriented picture processing
system designed for the PDP 11/45 computer. Its current
capabilities have enabled it to assume some of the computing
tasks in processing the NVL imagery data bases.

Figure 2 shows the basic hardware configuration
of the system. Picture files are stored locally on disk,
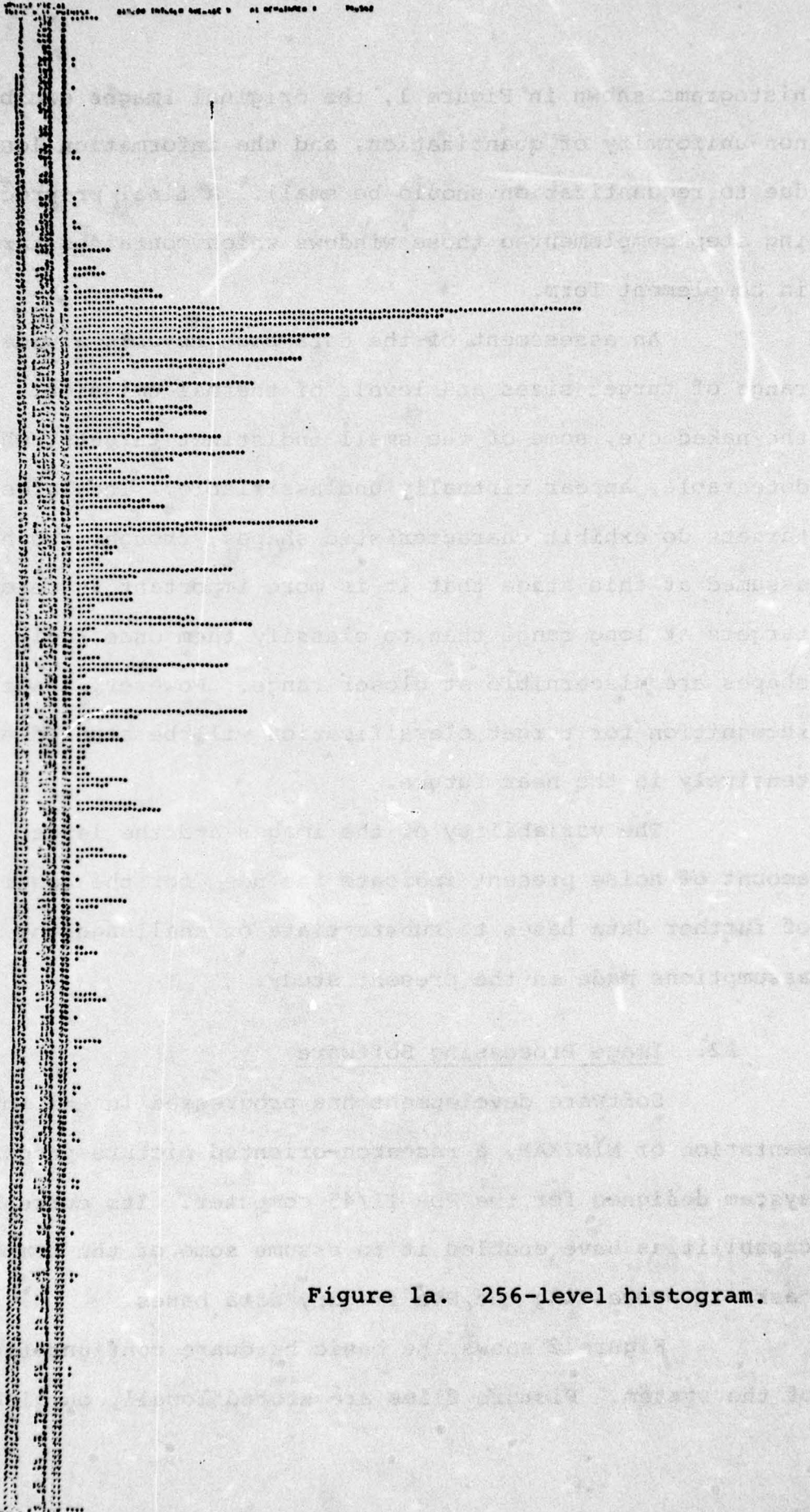
Figure 1a.   256-level histogram.
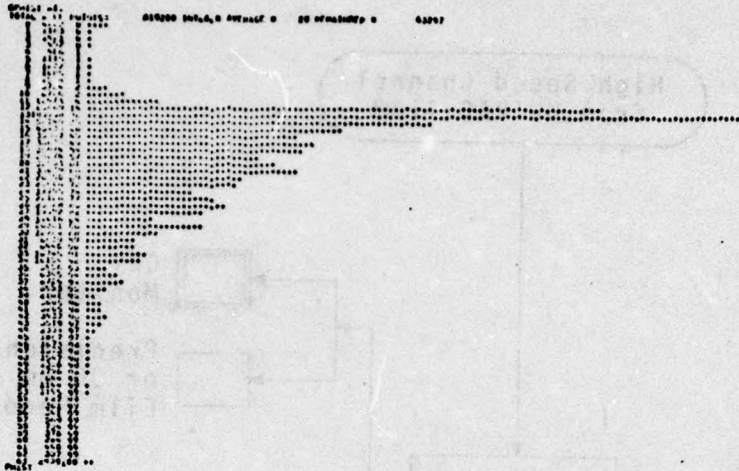
Figure 1b.   Same histogram re-
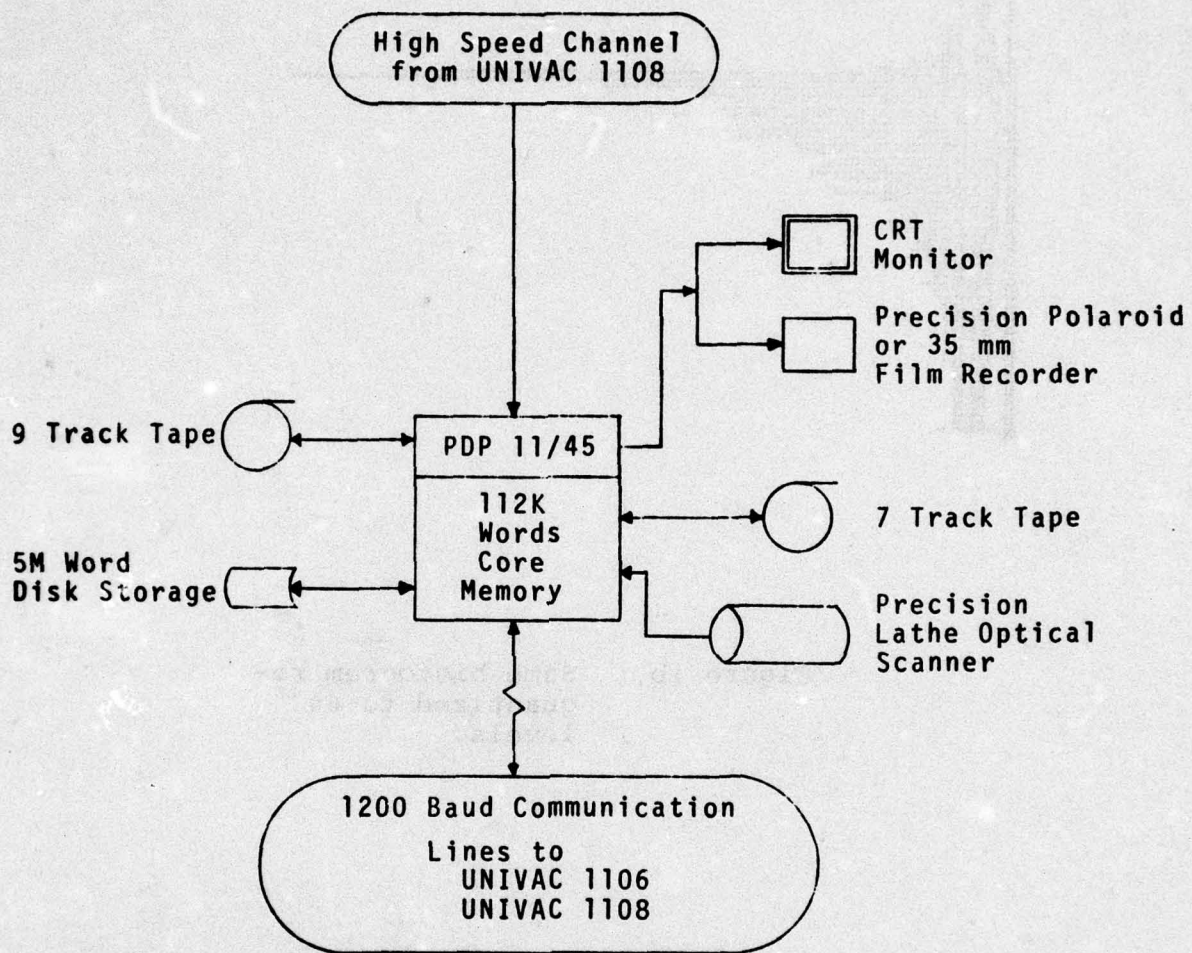   quantized to 64
              levels.

Figure 2: Picture Processing Hardware Configuration

9-track or 7-track tape. Picture data can be transferred to the PDP 11/45 from the mass storage facilities of a UNIVAC 1108 computer via a high-speed UNIVAC 1108 channel. Images may be input from a precision drum-type optical scanner, and output to either a CRT monitor or a precision Polaroid or 35mm film recorder. Medium speed communication lines to UNIVAC 1106 and UNIVAC 1108 computers provide additional paths for picture data transfer and for program development activities.

MINIXAP has been designed in a four-level hierarchy. The bottom level, written in PDP 11 assembly language, manages a data base of picture files and provides device-independent I/O of picture intensity data. The second level provides a convenient command interface to the bottom level from the programming language LISP. The third level, written in LISP, is a collection of picture processing algorithm skeletons. An algorithm skeleton is a program-like structure in which certain arguments and functions are left unspecified until the skeleton is prepared for execution. At that time, the appropriate arguments and functions are associated with the skeleton, and the completed program may be executed. Many common image operations are sufficiently similar that they may be regarded as instantiations of algorithm skeletons. Figure 3 illustrates an algorithm skeleton for a picture processing operation which uses one input picture, generates one output picture, and in which the transformation function is a neighborhood operator. The Roberts gradient operator is an example of a neighbor-
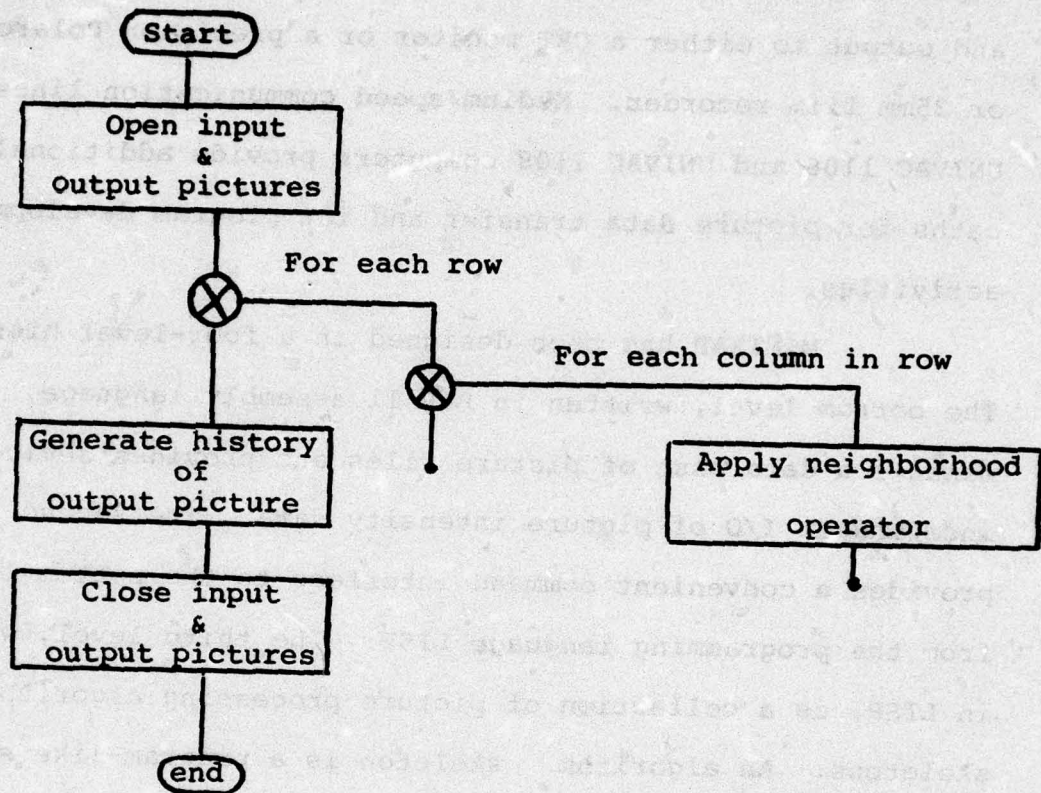
Figure 3.   A Picture Processing Algorithm
             Skeleton for Neighborhood Opera-
             tions.

hood operator using a 2x2 neighborhood size. The fourth level in the MINIXAP hierarchy is a collection of application packages, which use the algorithm skeletons to perform picture processing operations.

Because of the interactive nature of the user interface language LISP, the generality of the data structures afforded by LISP, and the collection of algorithm skeletons provided, MINIXAP should be a useful tool for use in picture processing algorithm development.

The following utility routines and application packages are currently available in MINIXAP:

(1) Utilities for:

   (a) picture printing

   (b) histogram generation and printing

   (c) picture copying

   (d) filename manipulation to facilitate the handling of large data bases

(2) Application packages

   (a) a cooccurrence matrix generator

   (b) an edge detection package, containing the Roberts gradient, Laplacian, "3 by 3" gradient, and "DIFF" operators

   (c) a propagation package, containing shrink/expand routines, thinning operators, distance transform and skeletonization operators, and border-following routines

   (d) a picture compression package for block averaging

Thus far, MINIXAP has been used in the object windowing, automatic threshold determination and noise cleaning phases of processing of FLIR imagery. It is anticipated that much of the future image processing algorithm development and testing for FLIR imagery will be done using MINIXAP.

B. Models for FLIR Image Understanding

The purpose of an image model is to define and account for significant variables of an image processing problem situation. Such models can suggest or substantiate algorithmic techniques, predict critical parameter values such as thresholds, and provide performance measures. As an initial step, we have chosen to model one aspect of FLIR imagery based on the simplified assumption that targets appear as homogeneous "hot" regions within a homogeneous "cooler" surround. Operations which respond to edges by assigning high values also respond to homogeneity with low values. A model which describes the transition from background to object can be used to predict a threshold gray level for separating object from background. In future work we plan to investigate a model involving the projective geometry of the image, to be used in predicting object size and orientation.

The model presented in this section is basically a first approximation to the real-world situation, since it assumes that the target and background have essentially constant gray levels (except for noise), and that the edges between target and background are ramplike. A more realistic model would take into account gradations of gray level across the image (e.g., due to range or terrain slope differences), and would treat edges as smooth transitions. (Gradations across the image may be unimportant when one processes relatively small windows, but could not be ignored

when analyzing entire frames.)

In spite of these limitations, the model does qualitatively predict the statistical measurements made on real images. It constitutes a first step in the development of more accurate models that should provide quantitative fits to real-image data.

In scene analysis it is often required to segment an image into background and object, where an object is a light area embedded in a background of darker gray level (or vice versa). A simple segmentation procedure is to select a gray level threshold to discriminate the pixels; pixels with gray level higher than the threshold are mapped into the object class and the rest into the background class. The optimum threshold for discrimination, given by Bayes decision theory, is the one that satisfies

$$p(t|\omega_0)P(\omega_0) = p(t|\omega_1)P(\omega_1) \tag{1}$$

where t is the gray level threshold; ω_1 and ω_0 are the two classes (object and background respectively); $P(\cdot)$ and $p(\cdot)$ are a priori and conditional probabilities. When the component density parameters or the prior probabilities are unknown, the location of the valley between the two modes of the mixture density, corresponding to the two modes of the component densities (assuming they are unimodal and "well separated"), is chosen as the threshold. But often such a threshold cannot be readily derived from the mixture histogram.

In the following we develop a model for optimal threshold selection which takes more accurate account of image structure.

B1. The noise-free case in one dimension

Let Fig. 4 represent the spatial gray level variation of a one-dimensional image. The background and object have gray levels s_0 and s_1 and are spatially connected by a ramp edge. Let e be the output of an edge operator*. Assuming the edges on both sides of the object are equally steep, the joint (gray level, edge value) histogram $p(s,e)$ will be as shown in Fig. 5. The impulse functons at $(s_0,0)$ and $(s_1,0)$ correspond to the background and the object respectively, the strength of these impulse functions being proportional to the areas of the background and the object. At the edge, the output of the edge operator is maximum ($= e_m$) and is constant for all gray levels at the edge. For $s_0 < s < s_1$ $p(s,e_m)$ is constant. Let

L = total length of the image

$d = |s_1 - s_0|$ (see Fig. 2)

w = width of edge on either side of object (see Fig. 1)

and $e(i) \triangleq |s(i) - s(i-1)|$. Then

$$e_m = \frac{d}{w} \tag{2}$$

probability $P(e=e_m) = \frac{2w}{L}$

and $p(s,e_m) = \frac{P(e_m)}{d} = \frac{2w}{Ld}$, for $s_0 < s < s_1$ (3)

Substituting eqn. (2) in eqn. (3) we get

$$p(s,e_m) = \frac{2}{Le_m} \tag{4a}$$

---

*This output will be referred to here, generically, as the gradient.
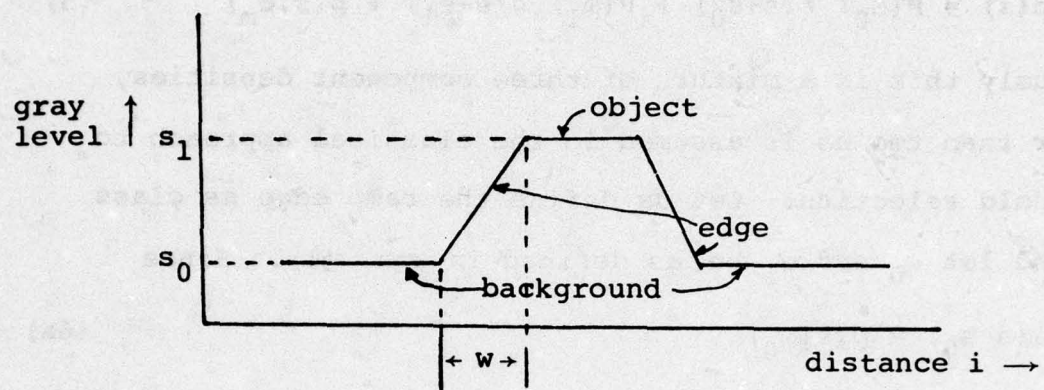
Fig. 4
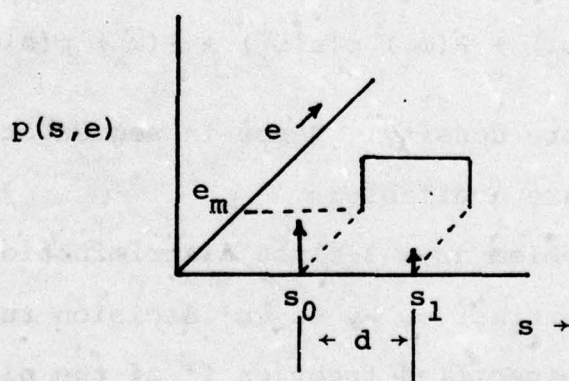
Fig. 5

Thus

$$p(s,e_m | s_0 < s < s_1) \propto \frac{1}{e_m} \qquad (4b)$$

where α denotes "is proportional to". Now the mixture density of gray level s is

$$p(s) = P(\omega_0)\, \delta(s-s_0) + P(\omega_1)\, \delta(s-s_1) + p(s,e_m) \qquad (5)$$

Obviously this is a mixture of three component densities, rather than two as is assumed in the classical approach to threshold selection. Let us define the ramp edge as class ω_2, and let ω_0 and ω_1 be as defined in eqn. (1). Since

$$\delta(s-s_0) = p(s|\omega_0) \qquad (6a)$$

$$\delta(s-s_1) = p(s|\omega_1) \qquad (6b)$$

and $\quad p(s,e_m) = p(s|e=e_m)\, P(e_m)$

$$= p(s|\omega_2)\, P(\omega_2) = \frac{1}{d} \cdot \frac{2w}{L} \qquad (6c)$$

eqn. (5) can be written as

$$p(s) = P(\omega_0)\delta(s-s_0) + P(\omega_1)\, \delta(s-s_1) + P(\omega_2)\, \frac{1}{d}$$

$$= P(\omega_0)\, p(s|\omega_0) + P(\omega_1)\, p(s|\omega_1) + P(\omega_2)\, p(s|\omega_2) \qquad (7)$$

which defines the mixture density. Hence in segmentation, the following choices are available:

(i)   Treat the problem as a 3-class discrimination problem and extract ω_1 by Bayes' decision rule.

(ii) Include an unspecified fraction f' of the pixels from ω_2 in ω_1 by selecting a threshold t_0, $s_0 < t_0 < s_1$, such that t_0 discriminates between ω_0 and ω_1 (at zero gradient)

(iii)   Select a threshold t_1 to include a specified frac-
tion f of pixels from ω_2 in ω_1.

In choice (i) the Bayes decision rule gives a set of piece-
wise linear discriminant functions. In particular the de-
cision rule is:

$$s < t_s, \; e < t_e \Rightarrow (s,e) \in \omega_0 \tag{8a}$$

$$s > t_s, \; e < t_e \Rightarrow (s,e) \in \omega_1 \tag{8b}$$

$$e > t_e \qquad\qquad \Rightarrow (s,e) \in \omega_2 \tag{8c}$$

where t_s is any threshold satisfying $s_0 < t_s < s_1$, and t_e is
any threshold satisfying $0 < t_e < e_m$. The above decision
rule is given by

$$(s,e) \in \omega_k \text{ if } \max_i \; p(s,e|\omega_i)P(\omega_i) = k. \tag{9}$$

The three discriminant functions are shown in Fig. 6. In
choice (ii) the space $\Omega = \{(s,e)\}$ may first be classified
into ω_2 and $\bar{\omega}_2$ by selecting a threshold t_e', $0 < t_e' < e_m$. The
set $\bar{\omega}_2$ may then be classified into ω_0 and ω_1 by selecting a
threshold t_0, $s_0 < t_0 < s_1$; however, in the final decision
rule the discriminant $s = t_0$ is extended to all gradient
values (e) and classifies the entire space Ω into only two
classes: ω_1 and $\bar{\omega}_1$; that is, the decision rule is

$$s < t_0 \Rightarrow (s_i) \in \bar{\omega}_1 \tag{10a}$$

$$s > t_0 \Rightarrow (s_i) \in \omega_1 \tag{10b}$$

Thus the pixels $\{(s,e)\} \,|> t_0, \; e = e_m\}$ are included in ω_1.
Essentially the threshold t_s is a Bayes classifier for all

$$p(s,e)$$
e
ω_2
e_m
$e = t_e$
s_0 ω_0 $s=t_s$ s_1 ω_1 $s \rightarrow$

Fig. 6

$$p(s,e)$$
e
$\bar{\omega}_1$
e_m
ω_1
$t_e \rightarrow 0^+$
s_0 t_0 s_1 $s \rightarrow$
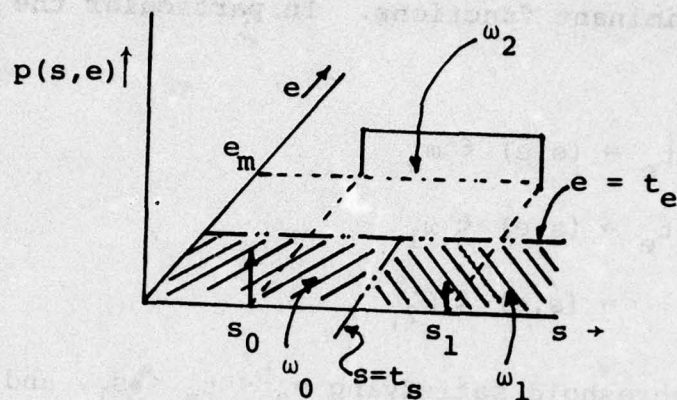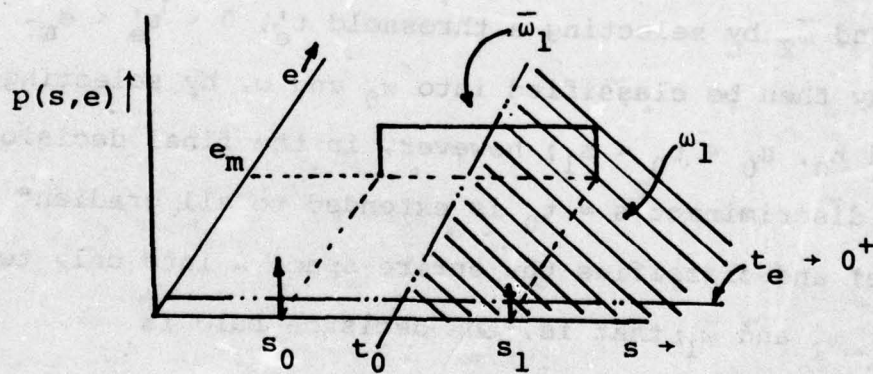
Fig. 7

pixels with zero gradient. Hence, the threshold t_e' can be chosen arbitrarily close to zero, i.e., $t_e' \to 0^+$. Thus in choice (ii) the segmentation procedure is to select a threshold t_0 that optimally classifies all pixels with gradient less than 0^+ into two classes, ω_0 and ω_1, and then extend the threshold to all gradients as in eqn. (10) above. The fraction f' of pixels from ω_2 included in ω_1 can be easily shown to be $f' = \frac{s_1 - t_0}{d}$. The discriminant function is shown in Fig. 7. It may be noted that the threshold t_e' would satisfy the same Bayesian optimality criterion as t_e in choice (i), and t_0 would satisfy the same criterion as t_s. In other words, the decision rule (8) will remain unchanged if t_e is replaced by t_e' and t_s is replaced by t_0. In choice (iii) a threshold t_e'' is selected to classify Ω into ω_2 and $\overline{\omega}_2$. Then the threshold t_1 is determined such that

$$P_r(s > t_1 | \omega_2) = f.$$

Now t_1 is used as a gray level threshold to partition the space Ω into ω_1 and $\overline{\omega}_1$ according to the rule:

$$s > t_1 \Rightarrow (s, \cdot) \in \omega_1$$

$$s < t_1 \Rightarrow (s, \cdot) \in \overline{\omega}_1 .$$

If the class conditional density of ω_2 is symmetric then $f = 0.5$ gives t_1 as the class conditional mean of ω_2. As a variation of choice (iii) one may select t_1 as the class conditional mean regardless of the shape of the class conditional density of ω_2. In this case every point in the edge between the object and the background is treated as a

potential candidate for the threshold and the actual

threshold selected is the mean value of all such candidate

thresholds. Thus the threshold t_1 is given by

$$t = E[s|\omega_2]$$

$$= E[s|e_m], \tag{11a}$$

Hence $$t = \int_{-\infty}^{\infty} s \, p(s|e_m) \, ds$$

$$= \frac{1}{d} \int_{s_0}^{s_1} s \, ds$$

$$= \frac{1}{s_1 - s_0} \cdot \frac{s_1^2 - s_0^2}{2}$$

$$= \frac{s_1 + s_0}{2} \tag{11b}$$

which is not surprising. Clearly, this suggests a method for

selecting the threshold: choose the points where the output

of the edge operator is high ($e = e_m$ in the example); the

mean gray level of such points gives the threshold.

B2.   Extension to two dimensions

In extending the simple case of Fig. 1 to two-
dimensional space the following assumptions are made: the
object is of constant gray level s_1 and is convex, the back-
ground is of constand gray level s_0, and at the edge of the
object gray levels increase from s_0 at the background to s_1
at the object monotonically and at a constant rate (see
Fig. 8).

The structure of the joint histogram of (s,e) in this
case remains basically the same as in the one-dimensional
case with one major exception. If two constant gray level
contours c_0 and c_1 are drawn through the points in the edge
region, with the gray level of c_0 greater than that of c_1,
then because of the shape of the object, c_0 contains fewer
pixels than c_1 does. Three simple observations can be made
regarding these contours: the number of pixels in a con-
tour is proportional to the length of the contour; the length
of the contour monotonically increases as the distance of
the contour from the object (measured in a direction ortho-
gonal to the contour) increases; and the gray level of the
contour decreases linearly as the distance from the object
increases. Thus $p(s|e_m)$ is a monotonically decreasing func-
tion from s_0 to s_1. In the simplest case of circular object
shape (and circular contours) this function can be shown to
be linear, as shown in Fig. 9 . If the Laplacian is used as
the edge operator then instead of a monotonic function
$p(s|e_m)$ will be two delta functions at $s = s_0$ and $s = s_1$.

In describing the simple case of Fig. 1 to two-
dimensions, made the following assumptions are made: the

Fig. 8

$p(s,e)$
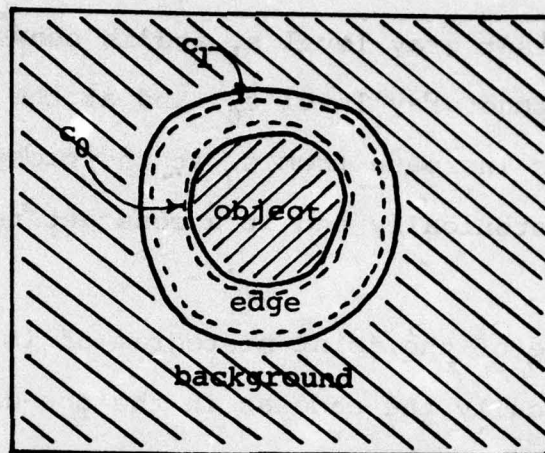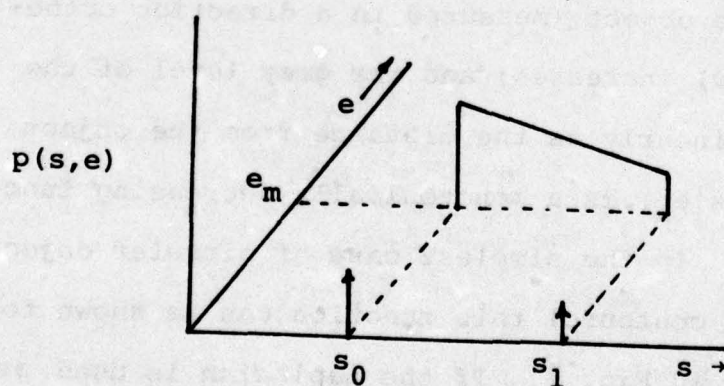
Fig. 9

However, in either case, the density $p(s,e)$ is still a mixture of three components and the three decision rules corresponding to the three choices still hold. Thus a threshold can still be selected by taking the conditional expectation of the gray level with the condition $e = e_m$.

B3. The Effect of Noise

Let the noise present in a scene be i.i.d (independent identically distributed) with zero mean normal distribution (variance $= \sigma^2$). The new gray level in the two-dimensional image space is

$$x(i,j) = s(i,j) + n(i,j) \tag{12}$$

where s is the original gray level as shown in Fig. 8 and Fig. 9, n is the normally distributed noise, and $x(i,j)$ is the gray level of the noisy image at (i,j). Clearly the noise is independent of the three classes ω_0, ω_1, and ω_2. Thus the components of the mixture density are given by the convolution of the noise density with the original component densities. Specifically

$$p(x|\omega_0) = p(s|\omega_0) * p(n)$$

$$\sim N(s_0, \sigma^2) \tag{13a}$$

$$p(x|\omega_1) = p(s|\omega_1) * p(n)$$

$$\sim N(s_1, \sigma^2) \tag{13b}$$

$$p(x|\omega_2) = p(s|\omega_2) * p(n) \tag{13c}$$

The density function for ω_2, unfortunately, is not so simple as that of ω_0 or ω_1. For the case shown in Fig. 9 the general shape of the component densities is as shown in Fig. 10. Thus it may not be feasible to select the threshold by locating the valley, in the mixture density, between the modes corresponding to those of $p(x|\omega_0)$ and $p(x|\omega_1)$.

To get some insight into the joint p.d.f. of gray level
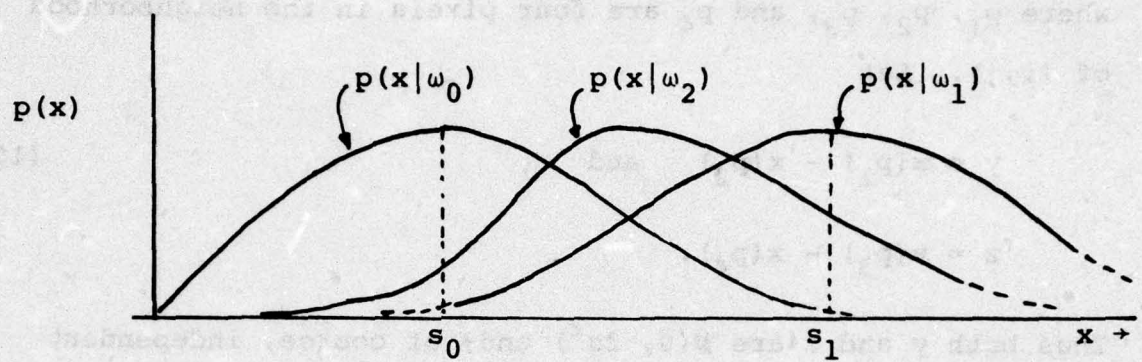
Fig. 10
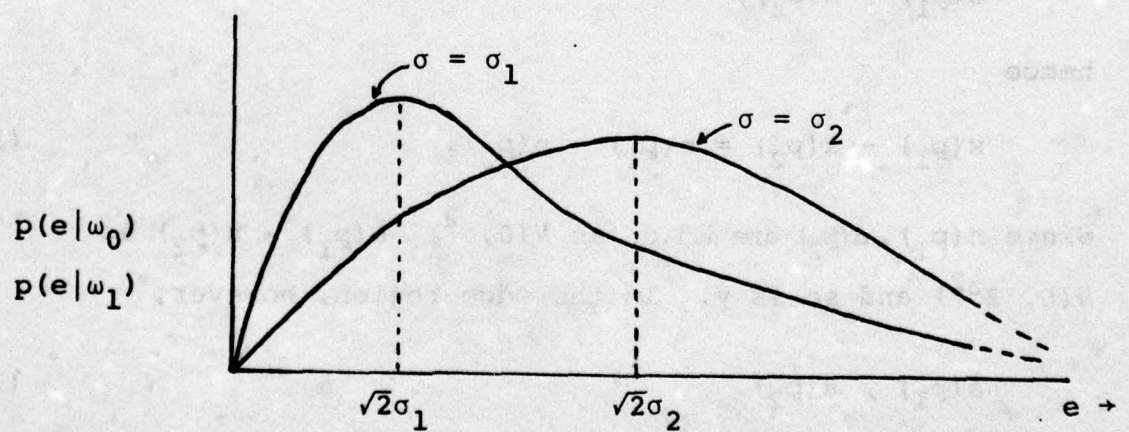
Fig. 11

and gradient let us assume that the edge operator is of the
form

$$e(i,j) = \sqrt{[x(p_1)-x(p_2)]^2 + [x(p_3)-x(p_4)]^2} \qquad (14)$$

where p_1, p_2, p_3, and p_4 are four pixels in the neighborhood
of (i,j). Let

$$y = x(p_1) - x(p_2) \quad \text{and} \qquad (15a)$$

$$z = x(p_3) - x(p_4).$$

Thus both y and z are $N(0, 2\sigma^2)$ and, of course, independent
in both the background and the object region. This is true
because

$$x(p_1) - x(p_2) = s(p_1) + n(p_1) - s(p_2) - n(p_2), \qquad (16)$$

and in the background as well as in the object region

$$s(p_1) = s(p_2), \qquad (17a)$$

hence

$$x(p_1) - x(p_2) = n(p_1) - n(p_2). \qquad (17b)$$

Since $n(p_1), n(p_2)$ are i.i.d. as $N(0,\sigma^2)$, $n(p_1) - n(p_2)$ is
$N(0, 2\sigma^2)$ and so is y. In the edge region, however,

$$s(p_1) \neq s(p_2) \qquad (18a)$$

$$s(p_3) \neq s(p_4). \qquad (18b)$$

Thus for ω_0 and ω_1

$$e(i,j) = \sqrt{y^2+z^2}$$

where y and z are independent $N(0,2\sigma^2)$. Therefore $e(i,j)$ is

Rayleigh distributed with p.d.f.

$$p(e|\omega_0) = p(e|\omega_1) = \frac{e}{2\sigma^2} \exp[\frac{e^2}{4\sigma^2}] \, u(e) \qquad (19)$$

where $u(\cdot)$ is the unit step function. The general shape of

the function is shown in Fig. 11 for $\sigma = \sigma_1$ and $\sigma = \sigma_2 > \sigma_1$.

The mean and variance of the gradient e in ω_0 and/or ω_1 are

easily computed as

$$E[e|\omega_0] = E[e|\omega_1] = \sqrt{\pi}\sigma \qquad (20a)$$

$$Var[e|\omega_0] = Var[e|\omega_1] = (4-\pi)\sigma^2. \qquad (20b)$$

Here a few observations are in order. First of all, the pre-

sence of noise has not only spread the gray level distribution

in the otherwise homogeneous region (object and background)

but has spread the gradient distribution also. Secondly, the

dispersion of the gradient in the otherwise homogeneous regions

is directly proportional to the noise dispersion. The mean

gradient in the homogeneous region also increases with the

noise dispersion. For the sake of tractability, assuming in-

dependence between gradient and gray level, the joint component

densities of ω_0 and ω_1 are given by the products of two

normal p.d.f.'s with Rayleigh p.d.f. Both the component den-

sities are unimodal in the bivariate space, the modes

occurring at $(s_0, \sqrt{2}\sigma_1)$ and $(s_1, \sqrt{2}\sigma_1)$ for ω_0 and ω_1, respec-

tively.

The case of the edge region, however, is complicated
by the inequalities (18). Assuming that in the neighborhood
of every point (i,j)

$$s(p_1) - s(p_2) = m \quad \text{and} \tag{21a}$$

$$s(p_3) - s(p_4) = n \tag{21b}$$

independent of location (i,j), then y and z of eqn. (15) be-
come $N(m, 2\sigma^2)$ and $N(n, 2\sigma^2)$, respectively. Hence in the edge
region the cumulative distribution function $P(e_1|\omega_2)$ is

$$\int_0^{e_1} \int_0^{2\pi} [\ \frac{1}{4\pi\sigma^2} \exp[-\frac{1}{4\sigma^2}((e\cos\theta-m)^2+(e\sin\theta-n)^2))]ede d\theta \tag{22a}$$

where $ede d\theta$ is the differential area in polar coordinate sys-
tem (e,θ) and the integrand in the square bracket is the joint
p.d.f. of (y,z) transformed into polar coordinates by

$$y = e\cos\theta \tag{22b}$$

$$z = e\sin\theta. \tag{22c}$$

The probability density function $p(e_1|\omega_2)$ is obtained by
differentiating expression (22a) w.r.t. e_1, which yields

$$p(e_1|\omega_2) = \frac{e}{4\pi\sigma^2} \exp[-\frac{1}{4\sigma^2}(e^2+m^2+n^2)] \cdot \tag{23}$$

$$\int_0^{2\pi} \exp[-\frac{1}{2\sigma^2}(m\cos\theta+n\sin\theta)]d\theta$$

In expression (22) and eqn. (23)

$$e_m = \sqrt{m^2+n^2}\ . \tag{24a}$$

Let us introduce a new variable ϕ defined by

$$\phi = \tan^{-1} \frac{m}{n} . \tag{24b}$$

In the edge region in the noise-free case ϕ gives the direction of gradient, where e_m gives its magnitude. Substituting eqn. (24) into eqn. (23) we get

$$p(e|\omega_2) = \frac{e}{4\pi\sigma^2} \exp[- \frac{1}{4\sigma^2}(e^2+e_m^2)] \int_0^{2\pi} \exp[- \frac{ee_m}{2\sigma^2} \sin(\phi+\theta)]d\theta$$

$$= \frac{e}{4\pi\sigma^2} \exp[- \frac{1}{4\sigma^2}(e^2+e_m^2)] \int_\phi^{2\pi+\phi} \exp[- \frac{ee_m}{2\sigma^2} \sin\theta]d\theta$$

Denoting the integral $\int_\phi^{2\pi+\phi} [\cdot]d\theta$ by $F(\phi)$ we have

$$p(e|\omega_2) = \frac{e}{4\pi\sigma^2} \exp[- \frac{1}{4\sigma^2} (e^2+e_m^2)] \ F(\phi). \tag{25}$$

When $e_m = 0$, eqn. (25) will reduce to eqn. (19).

Thus now the gradient in the edge region is no longer constant as it was in the noise-free case. The joint class conditional density of ω_2 is given by the product of eqn. (25) and eqn. (13c). The mode is a function not just of m and n as before, but also of σ^2, the noise variance. If the class conditional density in the noise-free case were uniform (as in Fig. 5) then the mode in the noisy case would be a straight line segment parallel to the x (gray level) axis.

In segmentation we still have the three choices, (i), (ii), and (iii), available to us just as in the noise-free

case, except that the corresponding decision rules must change. In choice (i) determining the thresholds corresponding to the Bayes decision was trivial in the noise-free case. For example, $p(e)$ is a mixture of two delta functions, at $e = e_m$ and $e = 0$, corresponding to $p(e|\omega_2)$ and $p(e|\bar{\omega}_2)$, respectively. Thus for all e in the range $0 < e < e_m$ we have $p(e|\omega_2) = 0$ and $p(e|\bar{\omega}_2) = 0$, and any such e is an optimal classifier; the problem is only to determine e_m from the noise-free sample picture(s), which is trivial. However, in the noisy case $p(e)$ is a mixture of p.d.f.'s given by eqn. (19) and eqn. (25) which may look as shown in Fig. 12. The Bayes discriminant function is satisfied by t_e (see Fig.12) where

$$p(t_e|\bar{\omega}_2) \; P(\bar{\omega}_2) = p(t_e|\omega_2) \; P(\omega_2); \tag{26}$$

but determining t_e now requires knowledge of the component p.d.f. parameter values and a priori probabilities. Hence t_e cannot be determined from the sample picture(s) alone. Instead one may locate the "valley" v_e (see Fig.12) in the mixture density (the mixture density can be estimated, e.g., by histogram $\hat{p}$, from sample picture(s)) and t_e can be estimated by v_e. Thus an alternative (for the noisy case) to the decision rule (8) is to seek valleys in the joint gray level and gradient histogram of the sample picture(s). Curves given by such valleys (see Fig. 13) can then be used as discriminants. In choice (ii), similarly, the threshold t_0 can be obtained by locating the valley v_0 in the conditional histogram $\hat{p}(x,e|e < t_e)$ where, as in the noise-free case,

$p(\cdot)$ ↑

$p(e)$

$p(e|\omega_2)P(\omega_2)$

$p(e|\bar{\omega}_2)P(\bar{\omega}_2)$

v_e t_e

$e \rightarrow$

Fig. 12

$\hat{p}(x,e)$ ↑

e

ω_0 ω_2 ω_1

possible discriminant by "valley seeking" method in choice (i)

Constant class conditional density contours

s_0 s_1 $x \rightarrow$
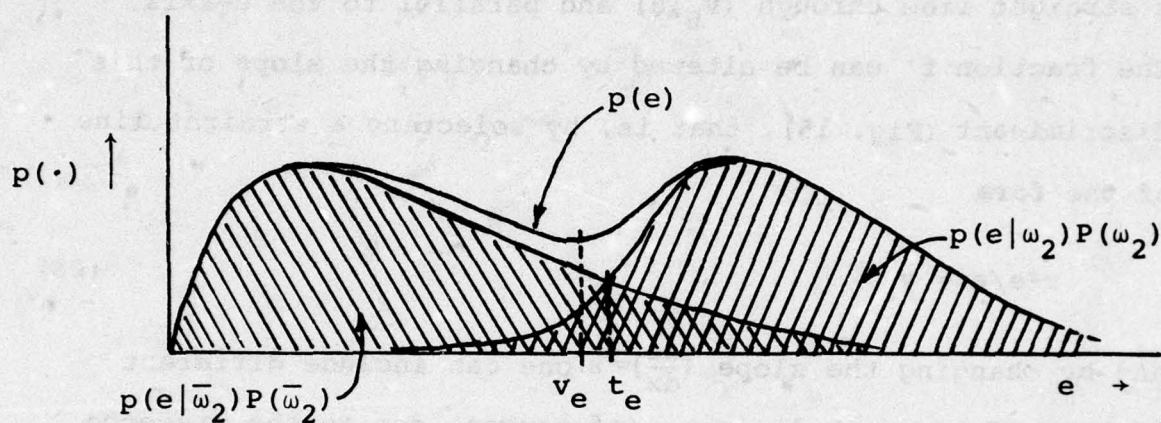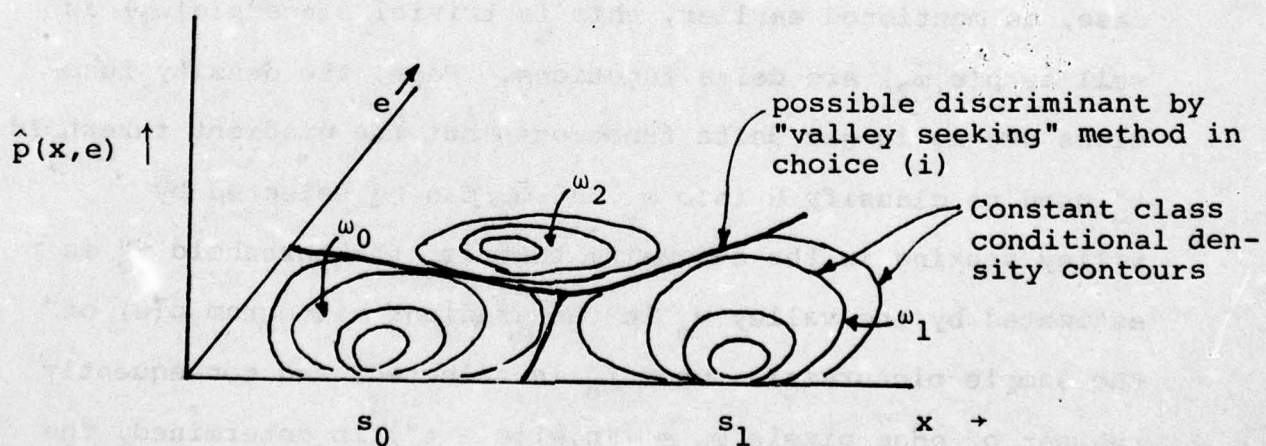
Fig. 13

$t_e \to 0^+$. The threshold t_0 is then extended to all gradient values to include an unspecified fraction f' of the edge pixels in ω_1 as shown in Fig. 14. The discriminant function, in this case, is given by

$$x = v_0, \tag{27}$$

a straight line through $(v_0,0)$ and parallel to the e-axis. The fraction f' can be altered by changing the slope of this discriminant (Fig. 15), that is, by selecting a straight line of the form

$$x-e/a = v_0 \tag{28}$$

and by changing the slope $(\frac{de}{dx}) = a$ one can include different numbers of edge pixels in ω_1 (of course, due to the presence of noise, this process includes some background pixels in ω_1). Section El uses this concept. In choice (iii) the threshold t_1 is selected from the class conditional density $p(x,e|\omega_2)$. This requires isolating ω_2 from $\bar{\omega}_2$ first. In the noise-free case, as mentioned earlier, this is trivial since $p(e|\omega_2)$ as well as $p(e|\bar{\omega}_2)$ are delta functions. Here, the density functions are no longer delta functions; but the gradient threshold t_e'', used to classify Ω into ω_2 and $\bar{\omega}_2$, can be selected by valley seeking in the e-domain; that is, the threshold t_e'' is estimated by the valley v_e in the gradient histogram $\hat{p}(e)$ of the sample picture(s). Once t_e'' is selected, and consequently the set of edge pixels $\omega_2 = \{(x,e)|e > t_e''\}$ is determined, the gray level threshold t_1 can be determined, as in the noise-free case, either by
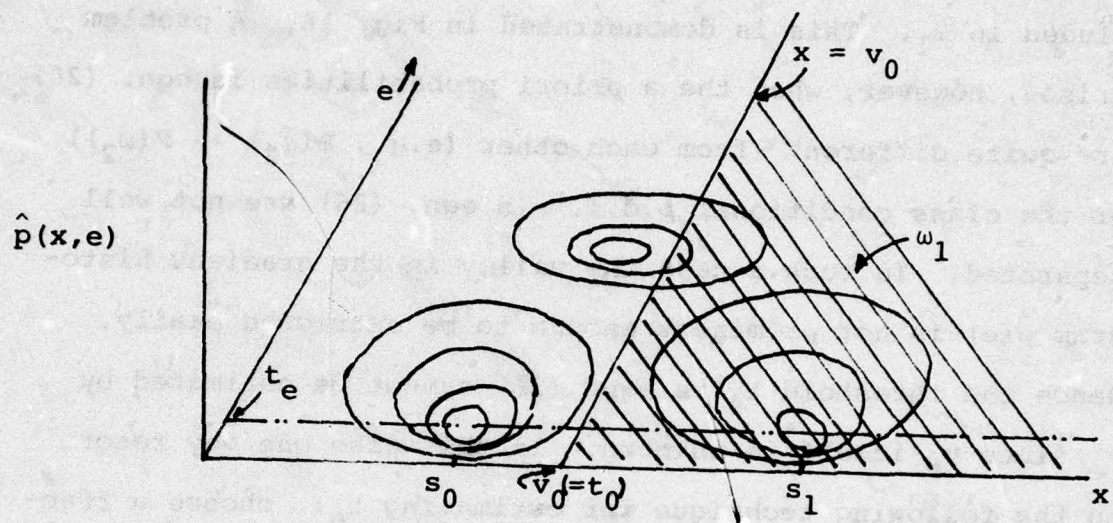
Fig. 14
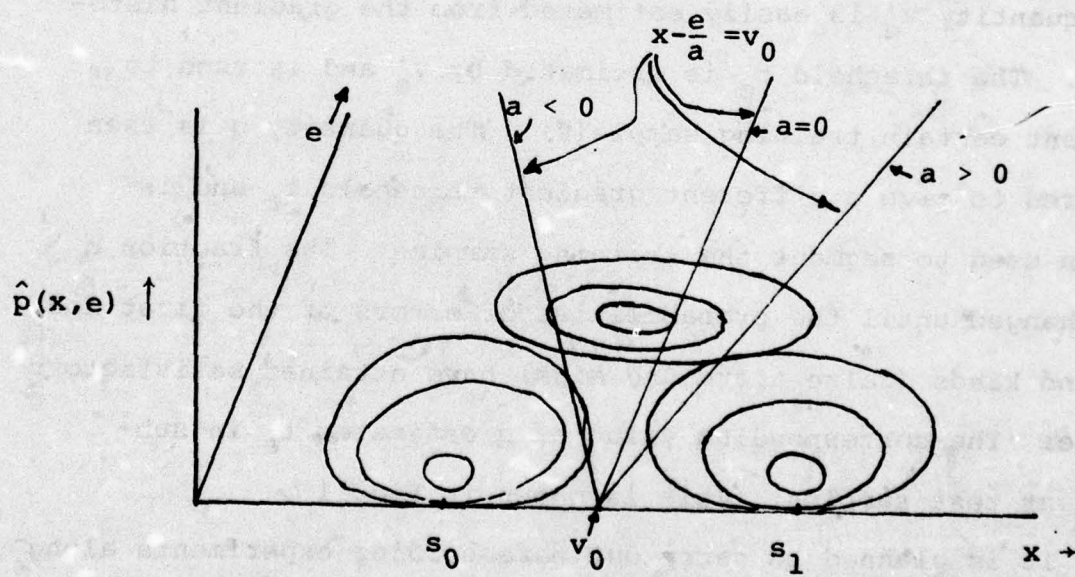
Fig. 15

$$Pr(x > t_1 | \omega_2) = f \qquad\qquad (29a)$$

or by

$$t = E[x | \omega_2] \qquad\qquad (29b)$$

where f is the specified fraction of edge pixels to be in-
cluded in ω_1. This is demonstrated in Fig. 16. A problem
arises, however, when the a priori probabilities in eqn. (26)
are quite different from each other (e.g., $P(\bar{\omega}_2) \gg P(\omega_2)$)
or the class conditional p.d.f.'s in eqn. (26) are not well
separated. In such a case the valley in the gradient histo-
gram $\hat{p}(e)$ is not prominent enough to be extracted easily.
Hence the threshold t_e in eqn. (26) cannot be estimated by
v_e since v_e itself is unknown. In this case one may resort
to the following technique for estimating t_e: choose a frac-
tion q and determine the point v_e' such that

$$Pr(e > v_e') = q. \qquad\qquad (30)$$

The quantity v_e' is easily estimated from the gradient histo-
gram. The threshold t_e is estimated by v_e' and is used to
segment certain training sample(s). The quantity q is then
altered to give a different gradient threshold t_e and is
again used to segment the training samples. The fraction q
is changed until the probabilities of errors of the first and
second kinds (false alarm and miss) have attained satisfactory
values. The corresponding value of q estimates t_e in sub-
sequent test samples. This is shown in Fig. 17.

It is planned to carry out thresholding experiments along
the lines suggested by this model (see Sections D and El for
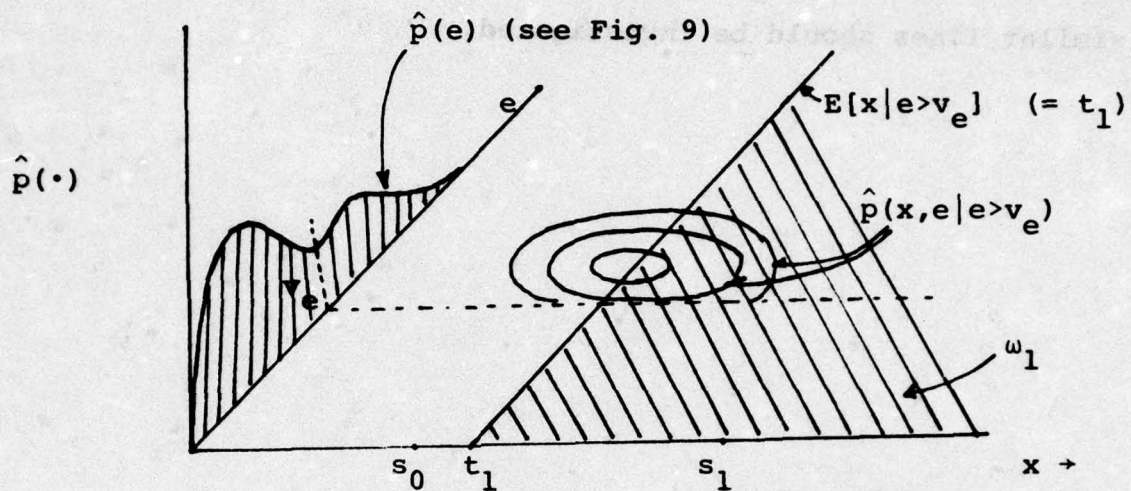
Fig. 16
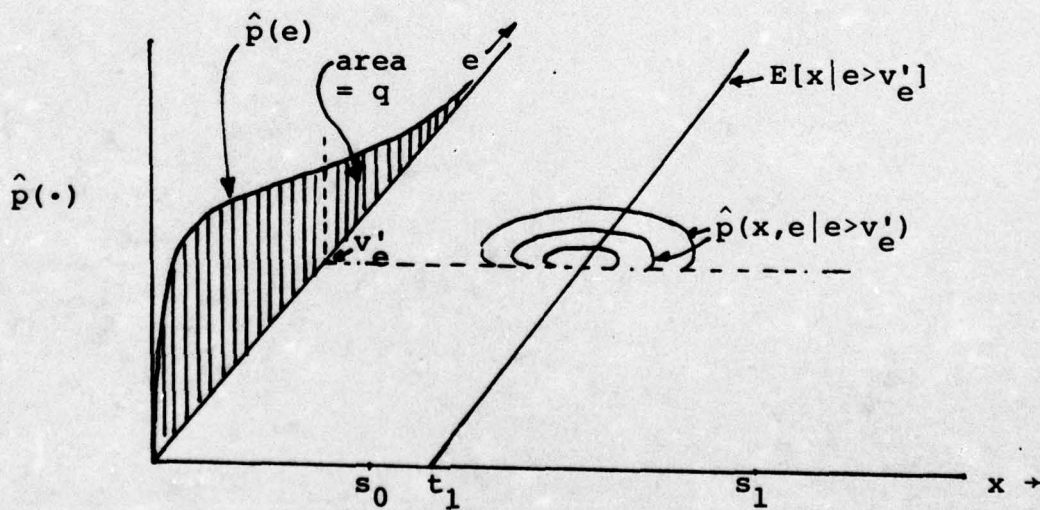
Fig. 17

some first steps in this direction), and to refine the model
as necessary. Meanwhile, it is felt that the model has pro-
vided important insights into the use of local image property
statistics in image segmentation. Other approaches along
similar lines should be investigated.
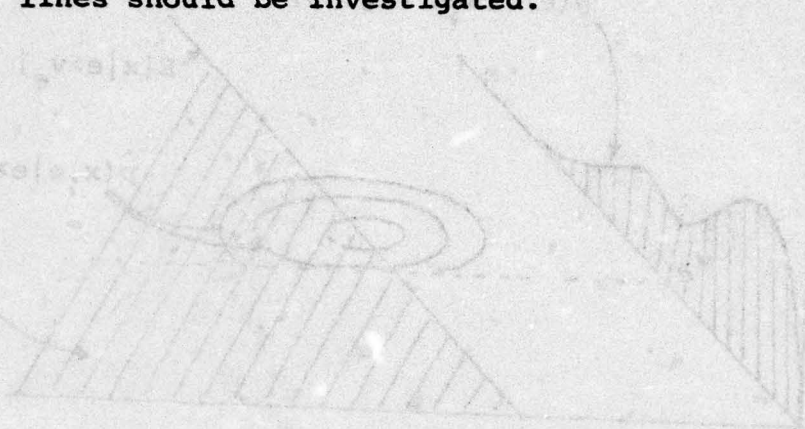
C.   Automatic Object Detection

Given an image divided into windows which may or may not contain targets, it is necessary to eliminate "noise" windows which contain no discernible objects. Subsequently, those windows containing objects can be analyzed to determine whether, in fact, the objects are targets.

A variety of techniques are available for testing whether a window is of sufficient interest to merit further processing. The a priori probability of occurrence of a "noise" window depends on the type of terrain being imaged. A hilly, wooded scene might, for example, contain many objects, while a flat desert scene might consist largely of empty windows. Terrain features such as hillocks and roads will also contribute objects for further analysis.

Later processing has the task of extracting the objects and classifying them as targets or non-targets based on shape features, gray level, size, texture, etc. This processing generally requires thresholding to segment the scene. Automatic threshold selection is treated in Section D. However, in the case of a noise window, choosing a threshold is not only futile but dangerous, in that segmentation, by its nature, will often find spurious "objects" in a noise window.

We are investigating methods of discovering noise windows based on the spatial distribution of gray levels in the window. Noise windows are generally homogeneous, and so the central moments evaluated on the whole window should

have high values, since there is no concentration of gray
level. On the other hand, windows containing objects should
have lower moment features. In Table 2 are displayed the
moment values for 10 noise windows and 30 target windows.
A classification experiment using the Fisher linear dis-
criminant produced the following confusion matrix:

	Classified as	
	Noise	Target
Noise	8	2
Target	11	19

A large number of the misclassified targets were small, in-
distinct, and located on uniform backgrounds. Inasmuch as
the misclassified targets were successfully thresholded
(see Section D), it seems likely that the statistical uni-
formity of 95% of the window obscured the hotter target
region.

A second experiment based not on the gray level
image but on the output of a difference operator (absolute
differences of 8x8 averages) showed that moment features
may respond better to clusters of edge values than to
clusters of gray values. In this experiment the confusion
matrix was $\begin{pmatrix} 8 & 2 \\ 5 & 25 \end{pmatrix}$. Here again, 4 of the 5 misclassified targets
were of the small indistinct variety. (See Tables 2c, 2d for
further details.)

Another experiment was performed to investigate the
SPAN technique [1, 2] as a method of detecting targets. This
technique (SPAN = Spatial Piecewise Approximation by
Neighborhoods) examines a set of neighborhoods of each

Image Ref. No.	Target Type	Average Gray Level	St. Dev. Gray Level	Centroid x	Centroid y	2nd Central Moments x²	y²	xy
3	T	22	5.97	32	34	1392	1526	-1.23
6	T	19	5.45	32	34	1368	1506	-6.41
11	T	22	5.59	32	33	1397	1459	-6.25
16	T	21	6.09	32	33	1395	1458	1.14
22	T	21	4.58	33	33	1429	1465	-.85
28	T	24	2.56	32	32	1421	1375	-.76
34	T	22	2.72	32	32	1365	1428	-2.55
45	T	23	4.73	33	34	1366	1520	2.62
52	T	19	2.79	32	33	1404	1463	-2.57
57	T	19	2.64	32	32	1388	1373	-10.88
3	R	22	5.59	33	34	1456	1542	.16
6	R	22	3.93	33	33	1425	1484	-3.50
22	R	22	3.67	32	33	1424	1427	-2.56
24	R	26	4.35	32	32	1353	1374	1.24
31	R	24	2.34	31	32	1352	1399	-8.71
34	R	23	2.28	32	32	1396	1421	-2.64
47	R	19	6.22	35	33	1558	1404	12.43
48	R	23	4.47	33	33	1429	1355	-3.46
55	R	21	1.98	32	32	1407	1414	.82
57	R	22	3.55	32	32	1400	1426	9.92
21	A	21	3.61	32	32	1400	1415	3.61
24	A	23	5.27	32	32	1411	1386	5.27
27	A	24	2.41	32	32	1387	1377	2.41
37	A	23	4.06	32	33	1384	1419	4.06
42	A	20	3.50	31	33	1356	1432	-.04
44	A	26	4.95	34	32	1527	1389	10.85
46	A	22	3.68	31	33	1327	1427	-.46
52	A	21	3.28	32	33	1377	1457	6.63
54	A	21	2.56	32	33	1409	1443	5.89
58	A	21	4.26	32	32	1392	1385	1.36
2	N	28	8.50	33	33	1451	1430	-3.06
8	N	28	5.83	31	34	1287	1496	-6.89
14	N	19	3.18	32	34	1386	1517	1.62
20	N	26	7.12	32	34	1408	1514	21.86
26	N	21	2.86	32	33	1418	1452	-3.34
32	N	21	1.89	32	32	1385	1398	.04
38	N	19	3.16	32	34	1395	1505	3.15
44	N	22	2.72	31	33	1351	1449	.22
50	N	21	2.55	32	32	1388	1369	-3.24
56	N	19	2.69	31	33	1355	1484	1.14

Table 2a. Feature values based on gray level for 30 target windows and 10 point windows.

Table 2b. Feature values based on 8x8 differences.

Image Ref. No.	Target Type	Average Gray Level	St. Dev. Gray Level	Centroid x	Centroid y	2nd Central Moments x²	y²	xy
3	T	2	2.76	28	23	145	128	-3.30
6	T	2	4.26	28	28	1508	1454	5.84
11	T	3	3.73	24	19	1164	166	-13.27
16	T	3	4.40	28	27	942	860	-.14
22	T	2	2.31	30	26	143	99	-14.36
28	T	1	1.29	24	32	111	168	-5.08
34	T	1	1.22	24	30	129	191	-11.76
45	T	2	2.50	24	34	194	2088	5.76
52	T	1	1.47	29	29	140	156	-31.80
57	T	0	.99	23	29	193	208	47.59
3	R	1	1.28	27	29	164	172	5.49
6	R	1	1.74	26	28	164	120	-20.46
22	R	1	1.41	34	18	98	117	20.73
24	R	2	1.74	29	23	146	142	-4.86
31	R	1	1.12	25	30	90	110	-1.05
34	R	0	.74	25	22	155	197	9.26
47	R	4	4.08	29	28	1225	693	-2.06
48	R	2	2.24	27	22	1617	123	5.65
55	R	0	.72	24	22	124	173	-17.50
57	R	1	1.03	28	25	196	194	7.39
21	A	1	2.10	28	19	139	92	3.36
24	A	1	2.27	28	16	133	83	-22.20
27	A	0	.97	23	32	126	141	40.08
37	A	2	2.27	27	25	126	135	12.26
42	A	1	1.95	21	18	117	129	16.73
44	A	2	1.71	28	28	165	164	1.34
46	A	2	1.89	24	23	97	151	18.50
52	A	0	.81	28	24	187	162	.81
54	A	0	.96	29	21	168	163	11.24
58	A	1	1.46	25	24	135	236	-16.75
2	N	0	.98	25	24	167	181	28.09
8	N	1	1.26	24	26	180	190	21.72
14	N	0	.76	22	28	225	160	-39.28
20	N	3	3.13	24	22	1087	1036	-14.07
26	N	0	.66	24	23	207	181	-13.57
32	N	0	.50	21	25	198	251	-20.08
38	N	0	.81	27	16	191	86	47.41
44	N	0	.45	27	17	197	139	-2.66
50	N	0	.87	24	26	191	164	-12.29
56	N	0	.70	29	18	172	152	-67.53

Feature	Fisher Direction	Misclassification Results	
		Target Image No.	Noise Image No.
Avg. G.L.	.316	3R	2
St. Dev. G.L.	-.946	6R	50
x^2	.00723	31R	
y^2	.0256	34T	
xy	.0706	34R	
		45T	
Threshold	30.5	52T	
		52A	
		54A	
		55R	
		57R	

Table 2c. Fisher linear discriminant classification experiment results using Table 2a data.

Feature	Fisher Direction	Misclassification Results	
		Target Image No.	Noise Image No.
Avg. G.L.	.199		8
St. Dev. G.L.	-.980	34R	38
x^2	.00113	48R	
y^2	.000562	52A	
xy	-.00350	55R	
		57R	
Threshold	-.630		

Table 2d. Fisher linear discriminat classification experiment results using Table 2b data.

image point, and picks the largest neighborhood that satisfies some uniformity criterion (see below). If this neighborhood is contained in some other point's largest uniform neighborhood, it is discarded. The result of this process is a set of irredundant, maximal uniform neighborhoods that provide an approximation to the given image. The technique is described in greater detail in [1, 2].

Figures 18 and 19 illustrate the SPAN technique and its application to two FLIR windows. The uniformity criterion was based on the chi-square test for normality. If the gray level distribution in a given neighborhood satisfied this test, the neighborhood was called uniform (in the sense that its gray level population was homogeneous). The maximal uniform neighborhoods determined in this way, shown in Figure 18, do separate the target and background regions, at least crudely. The method (as currently implemented) is computationally costly, but it deserves further study as a possible means of facilitating target detection.
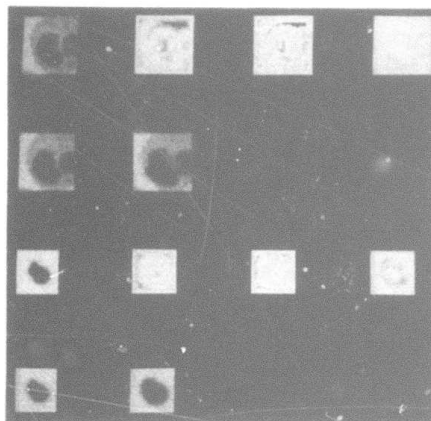
Figure 18a.   Chi-square Test for Unimodality.

Cell picture: Row 1.   Original image; maximal
                          SPAN radii at each point;
                          maximal radii with local
                          non-maxima suppressed;
                          detected edges.

                 Row 2.   Smoothed image; image re-
                          construction using SPAN

Chromosome: Row 3.   Same as Row 1.

                 Row 4.   Same as Row 2.
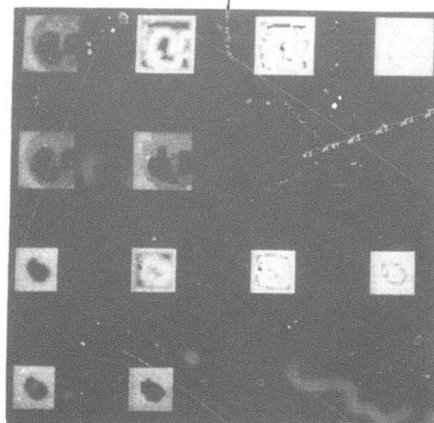
Figure 18b.   Multimodality Test.   Same as Fig. 18a.
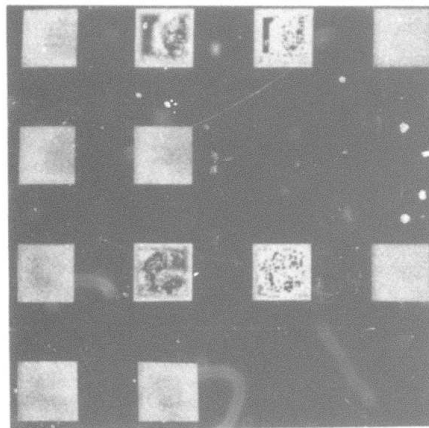
Figure 19a.   Chi-square Test.

Tank window: Same as Fig. 18a, Rows 1 and 2.

APC window: Same as Fig. 18a, Rows 3 and 4.
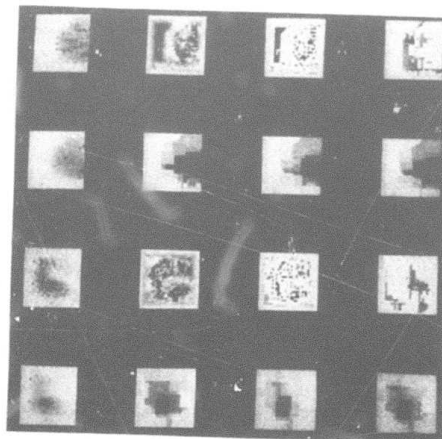
Figure 19b.   Same as Fig. 19a but with enhanced contrast. Additional reconstructions (Rows 2 and 4) using MAX, MIN and average recombination rules.

D. Automatic Threshold Selection

In Section B a model was proposed for images consisting of objects and background, each with characteristic gray level distributions. If the gray level histogram of the image is markedly bimodal, one may choose the threshold at the valley between the two peaks (possibly shifted towards the smaller peak when using a maximum likelihood estimate). However, as may be seen in Figure 20, the smaller the object, the less likely the histogram is to exhibit strong bimodality. The background distribution engulfs the object's gray level range and tests for bimodality are inconclusive.

Our approach [3] to solving this problem has been to select from the original image a set of points that are as likely to fall within the object as within the background. If one examines the output of operators which respond to edges, then high values should correspond to points falling at or near object edges. These points are as likely to lie on the object as on the background and their mean value should correspond to the desired threshold. At first it was thought that the distribution of such points for a sufficiently coarse detector would be bimodal. However, the model of the previous section has shown the distribution to be unimodal with a peak at the mean.

A number of edge operators were tried in connection with this approach. Figure 21 shows edge values for the following operators:

Figure 20. 156 Windows - sampled to 64 X 64.
Each window has a corresponding histogram in
which grid lines identify intervals of 100 image
points. Image reference numbers refer back to
ground truth.
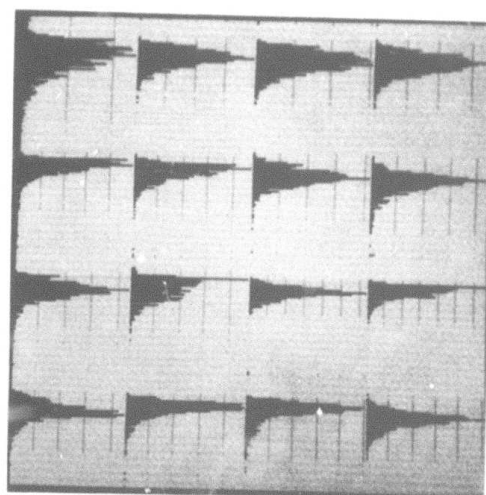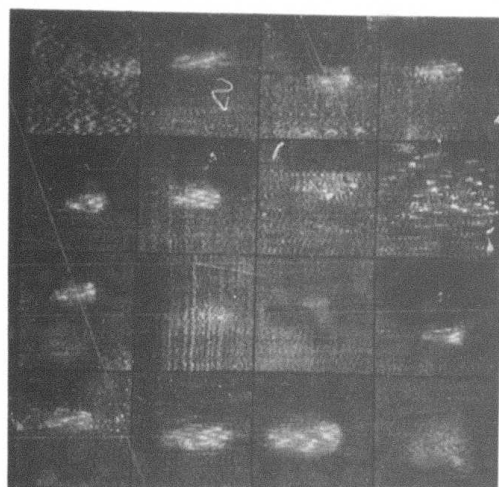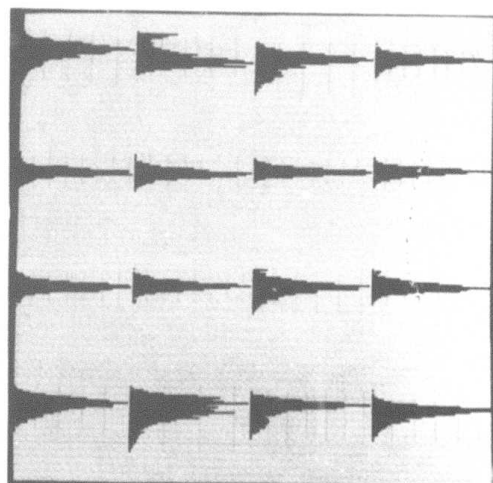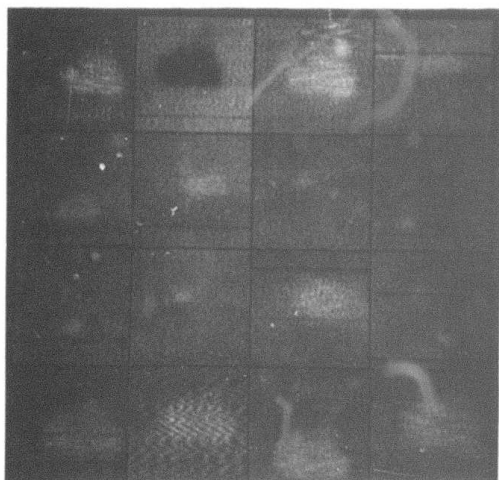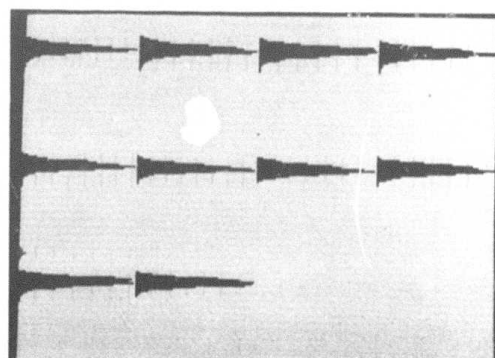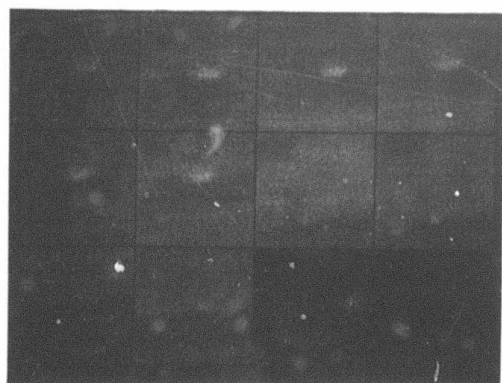
Figure 20a. 42 Tank windows, histograms and image reference numbers.
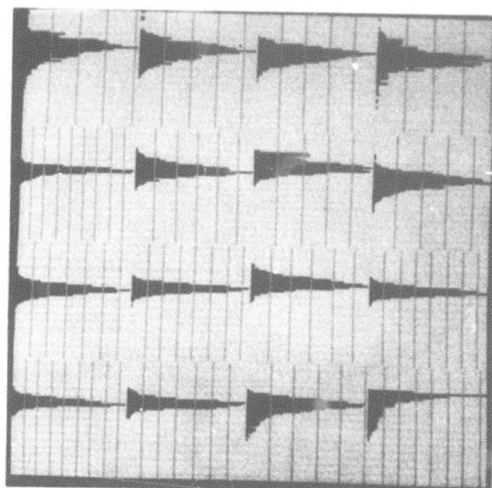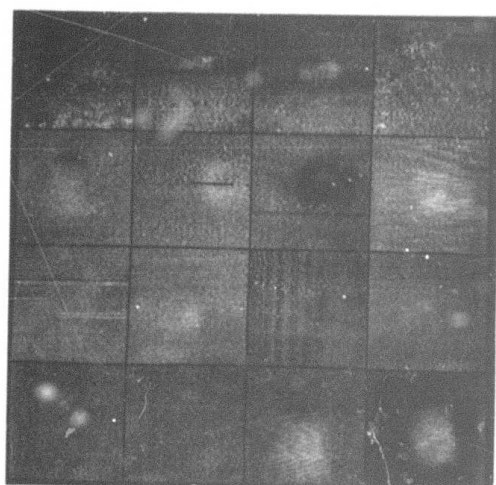
3R
4R
6R
9R
18R
22R
23R
24R
26R
31R
32R
33R
34R
35R
41R
47R
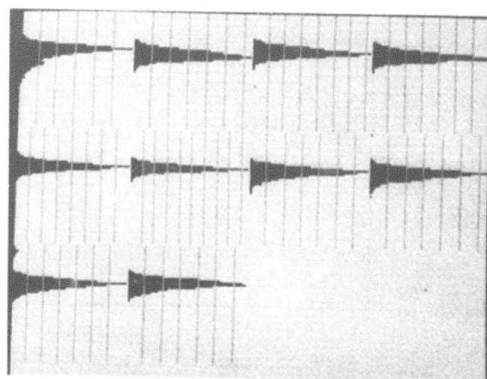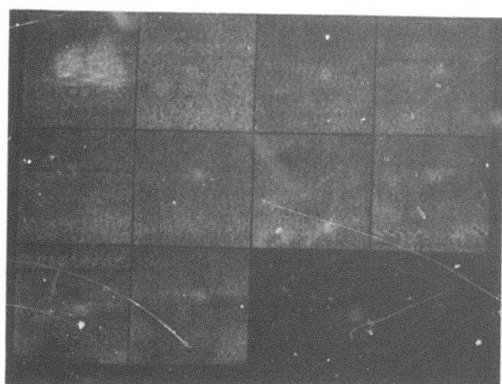
43R
51R
52R
53R
54R
55R
55R
57R
58R
59R

Figure 20b.   26 Truck windows, histograms and image ref. nos.
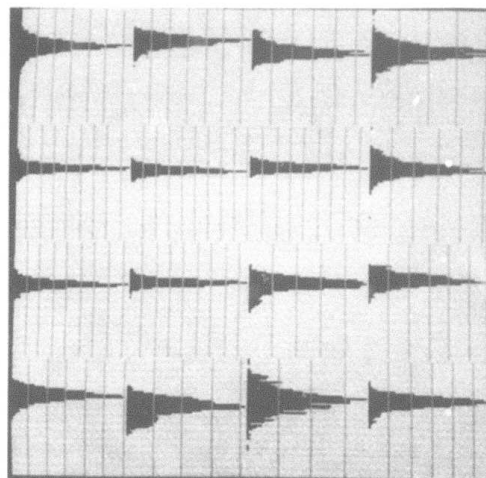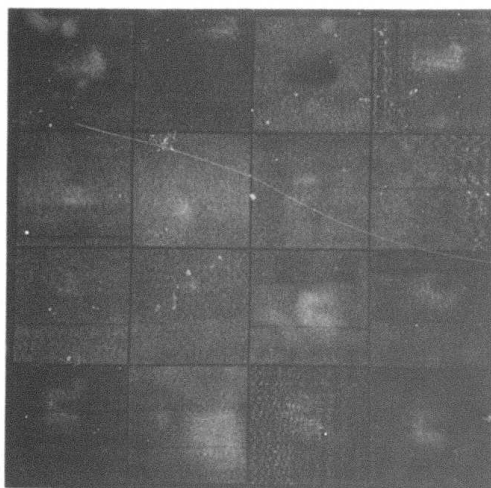
21A
22A
23A
24A
27A
28A
32A
33A
34A
35A
37A
38A
42A
44A
45A
46A
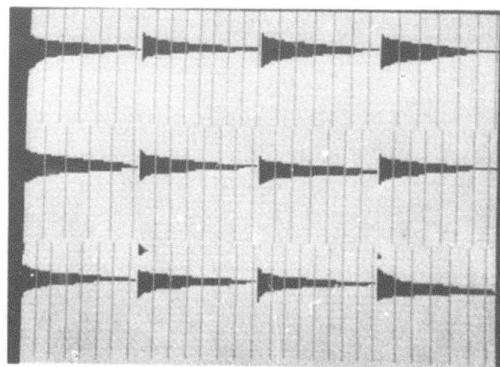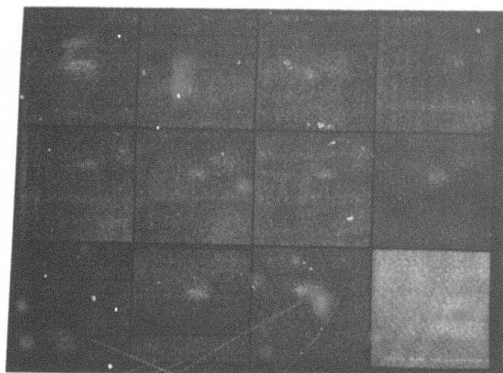
48A
50A
51A
52A
53A
54A
55A
56A
57A
58A
59A
60A

Figure 20c. 28 APC windows, histograms and image ref. nos.
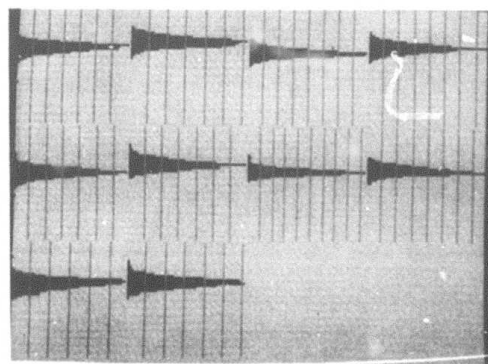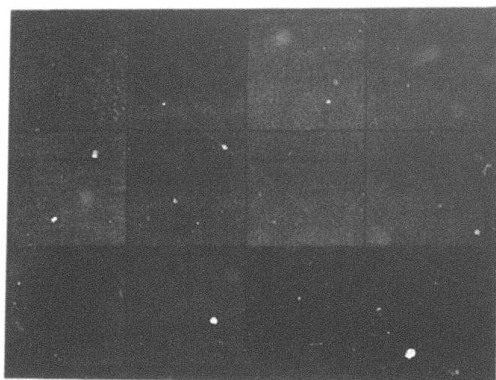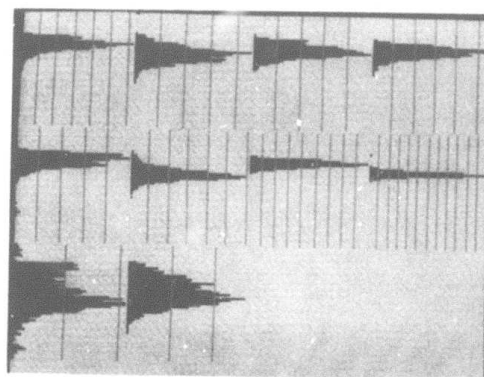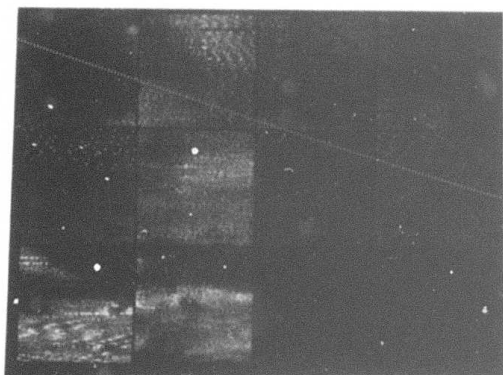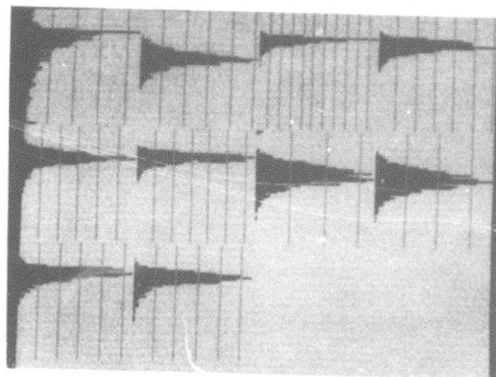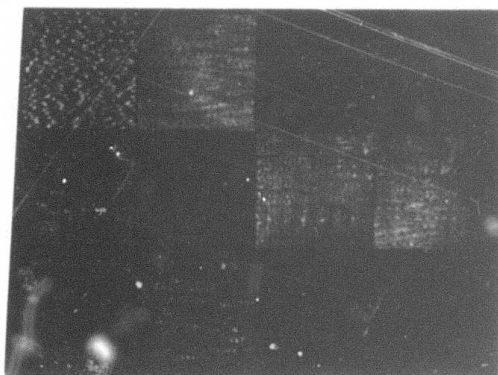
Figure 20d. 60 Noise windows and histograms. Image ref. nos.
are consecutive 1N - 60N.
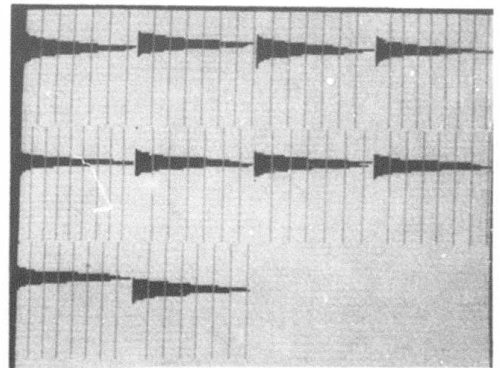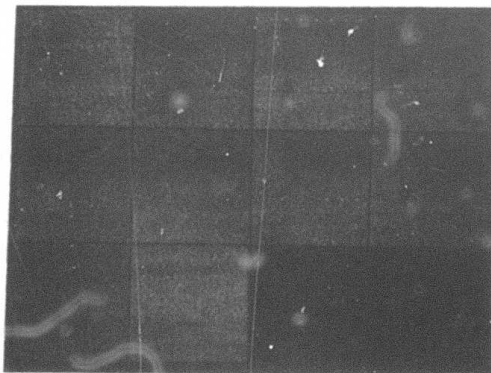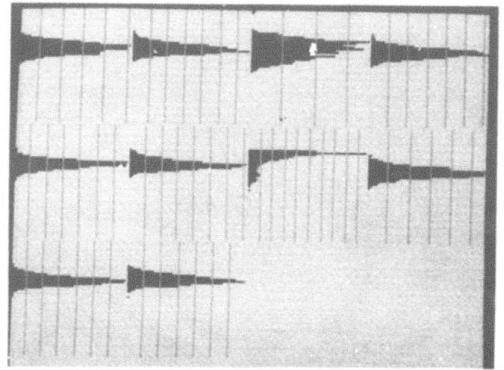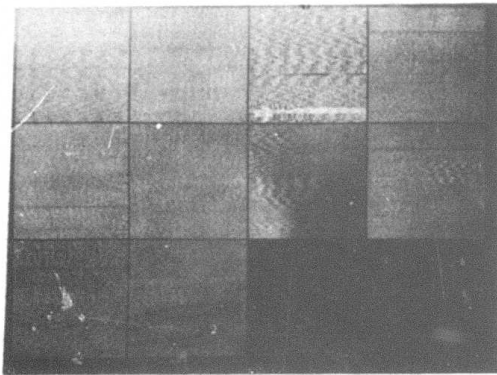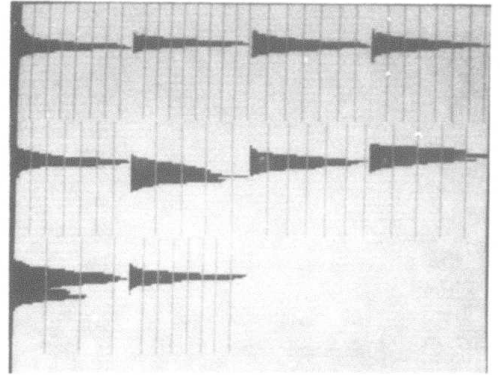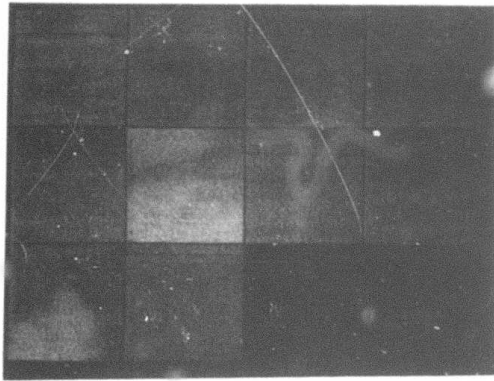
Figure 20d continued.

Figure 21. Edge values and above-threshold edge values.

Key:

Original image

	Edge values	Above-threshold edge values 80-file	90-file
Laplacian			
Roberts gradient			
3x3 gradient			

Original image

	Edge values	Above-threshold edge values 80-file	90-file	95-file
2x2 difference				
4x4 difference				
8x8 difference				

Key to images used: T = tank, R = truck, A = APC, N = noise
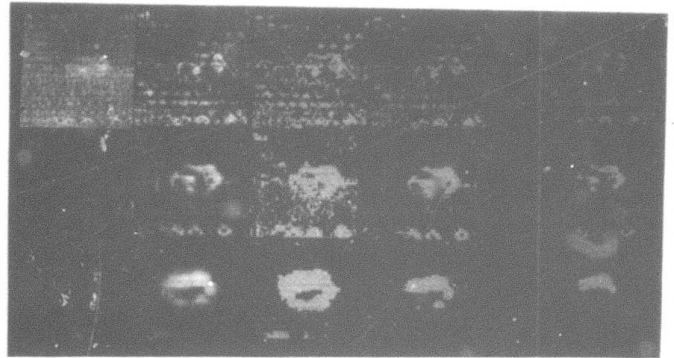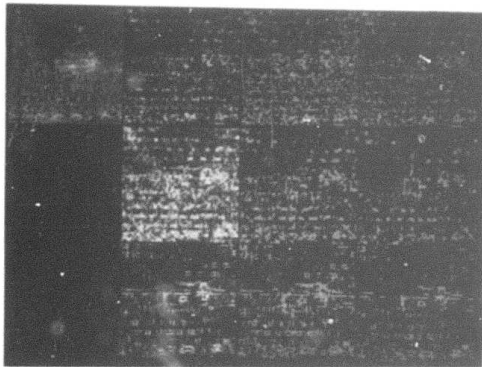
3T
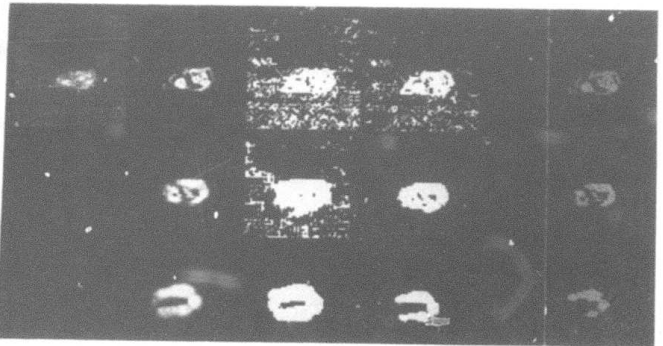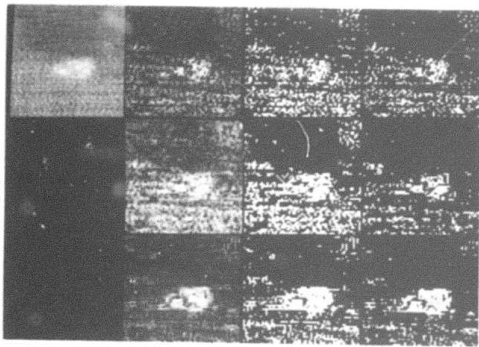
6T
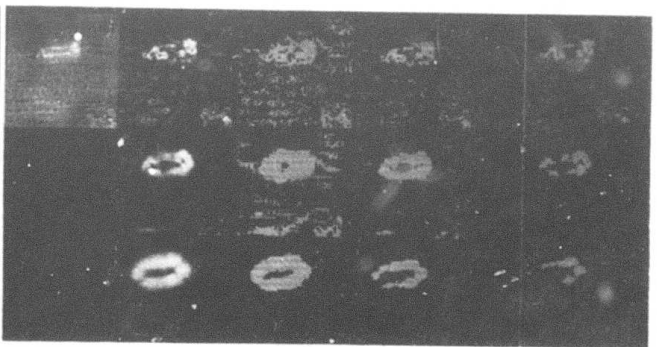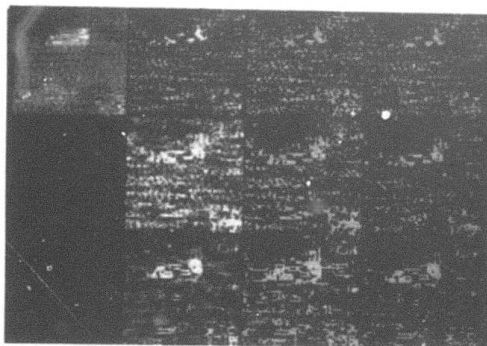
11T

Figure 21 (continued)
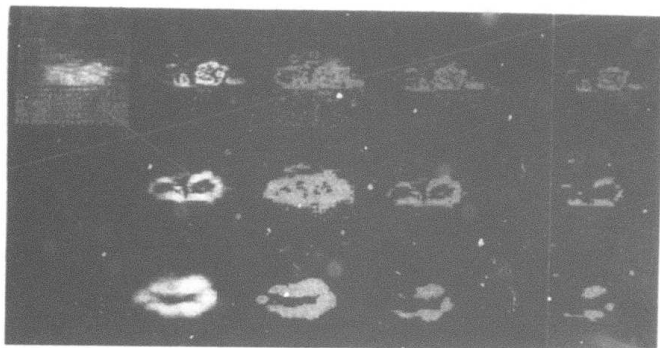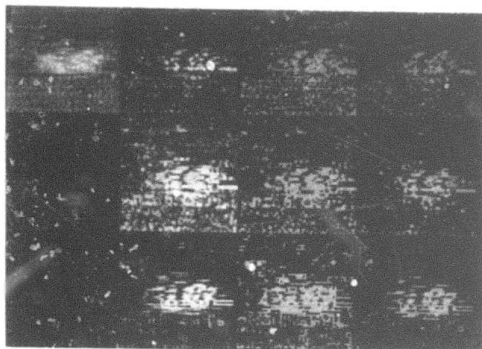
16T
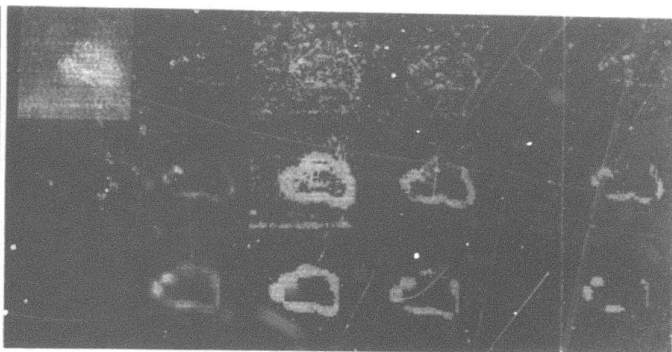
22T
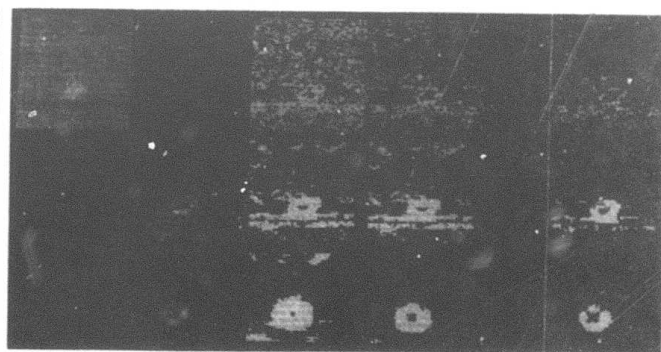
34T

Figure 21 (continued)
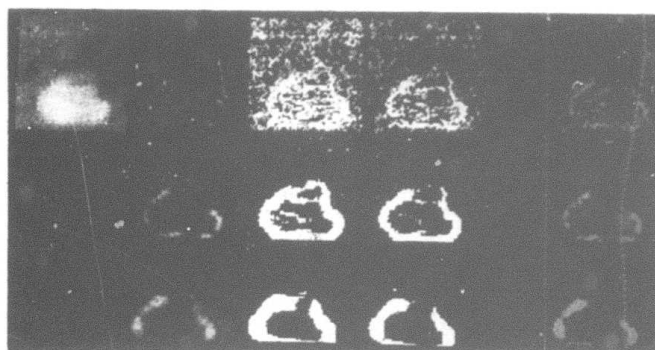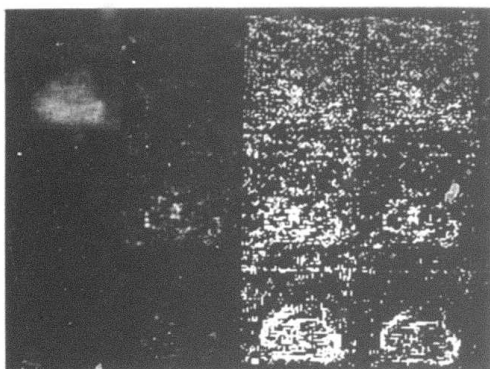
45T
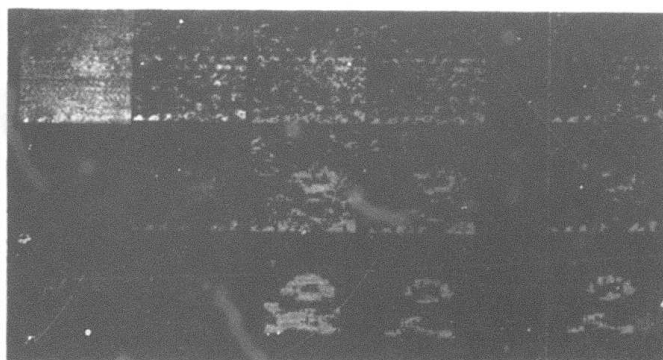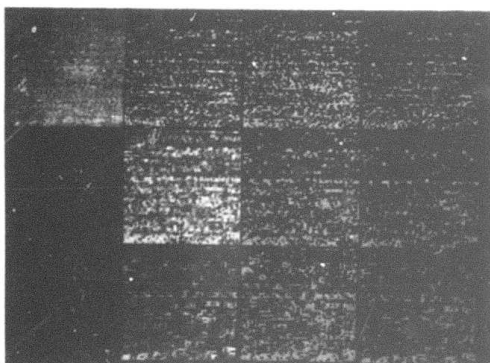
3R

6R

Figure 21 (continued)
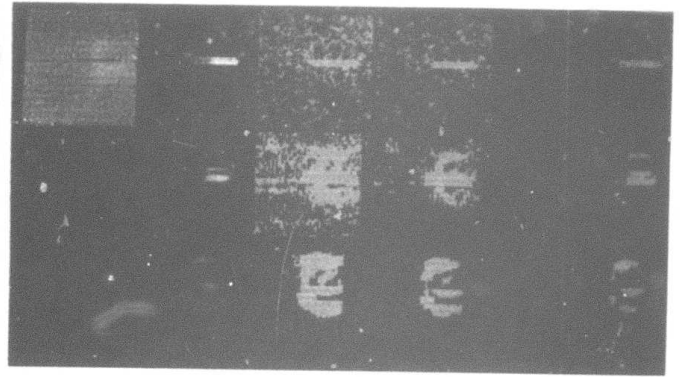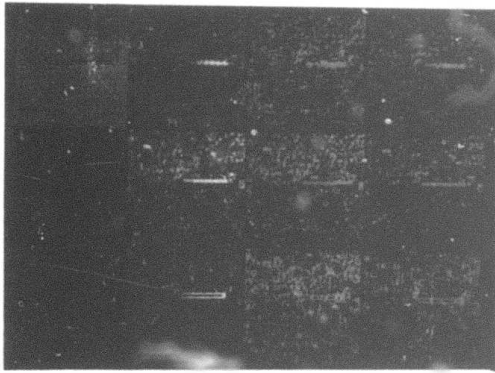
22R
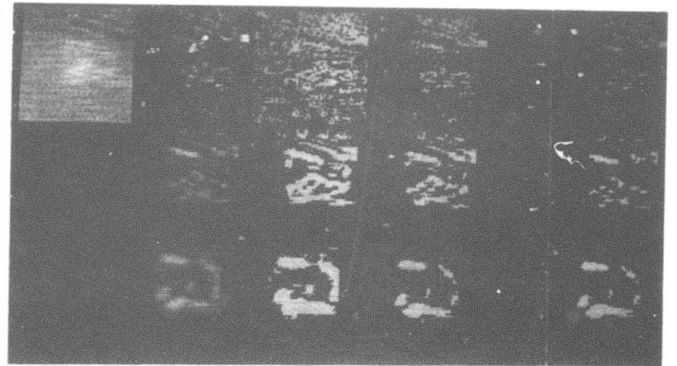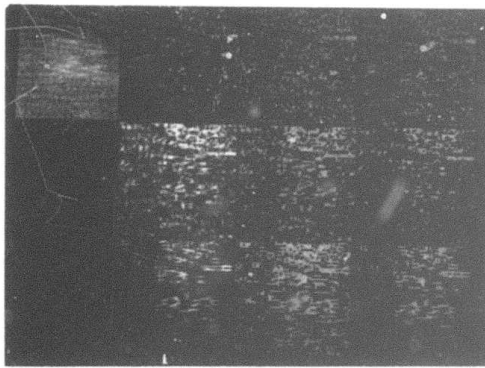
24R
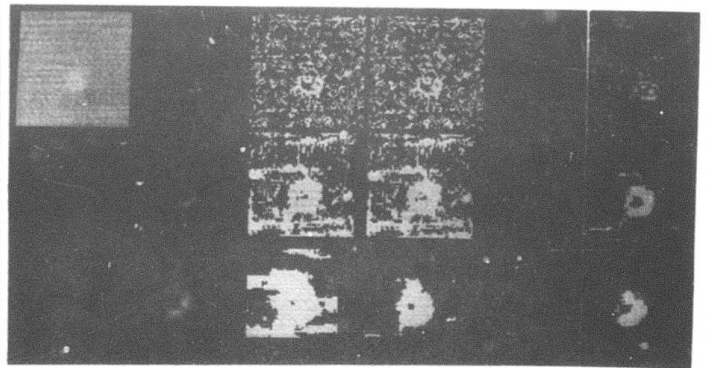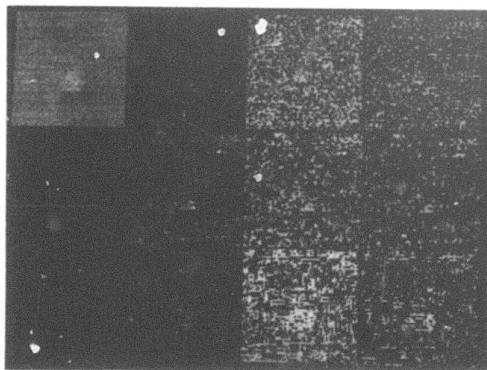
31R

Figure 21 (continued)
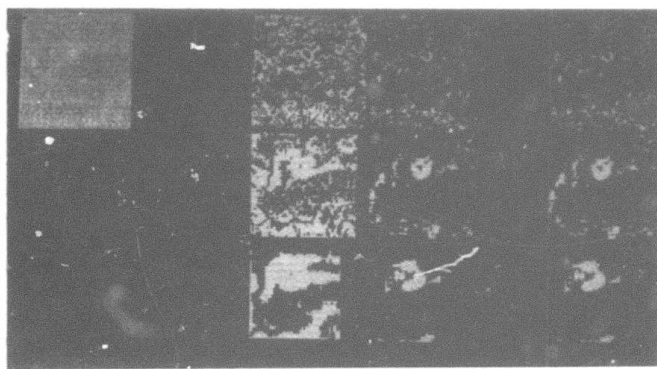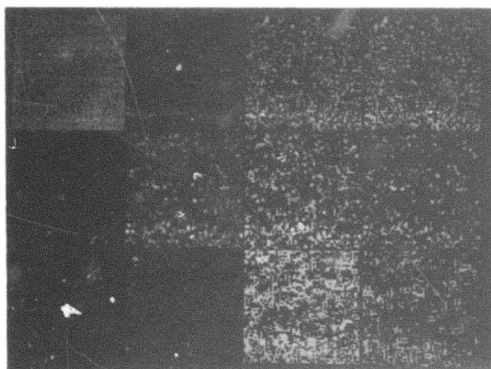
34R
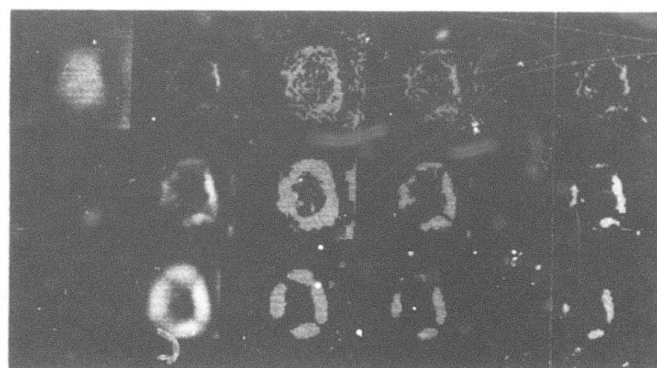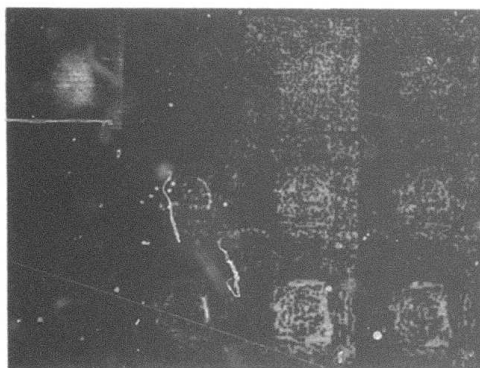
47R
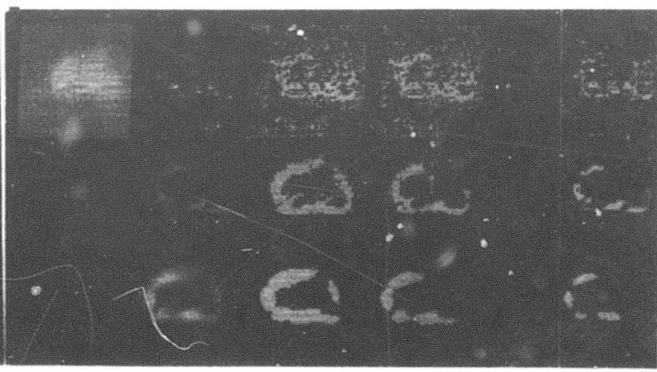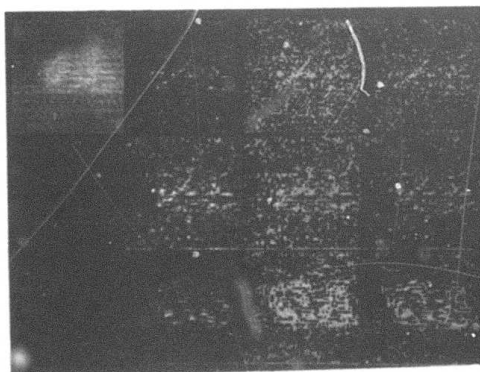
48R

Figure 21 (continued)

21A

24A

27A
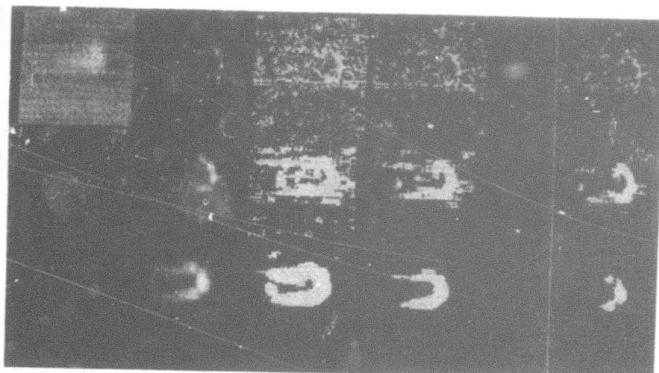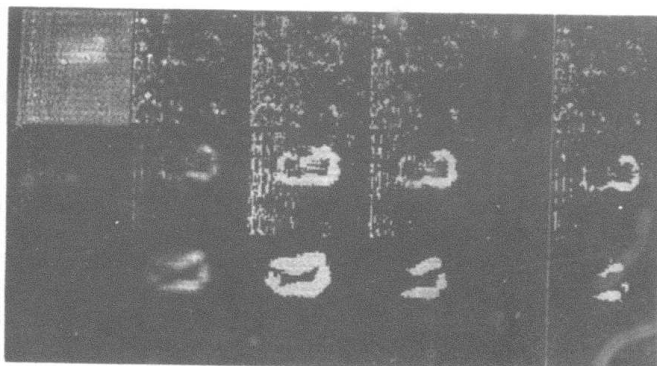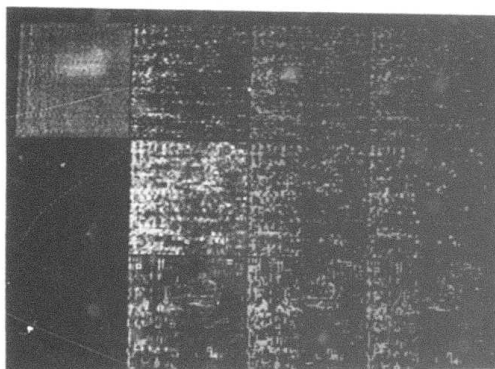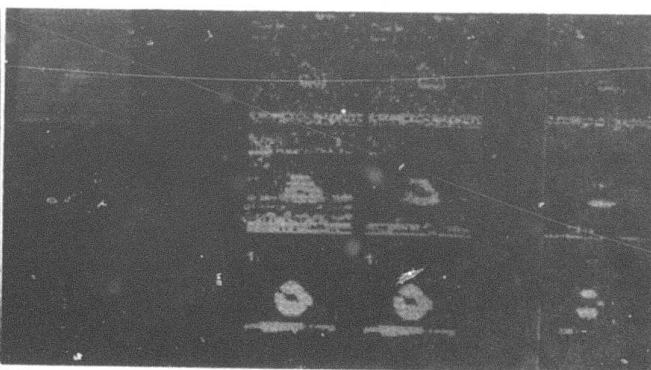
Figure 21 (continued)
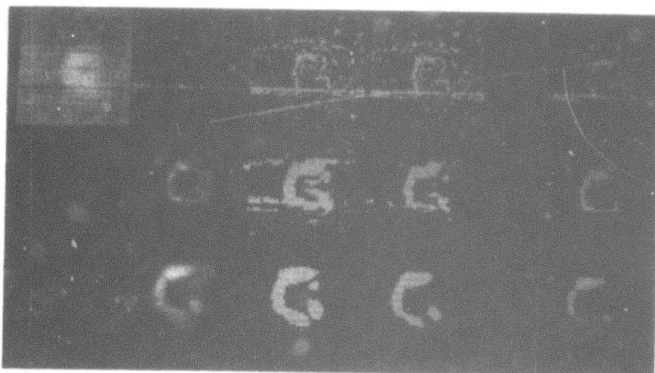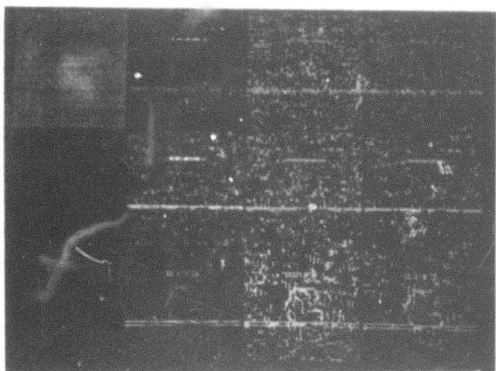
37A
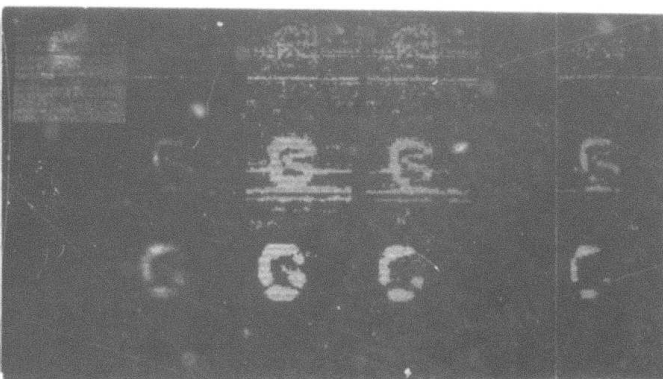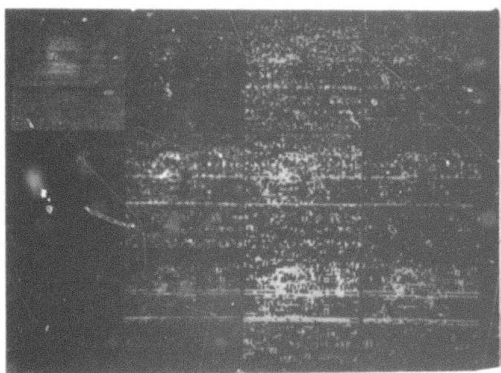
42A
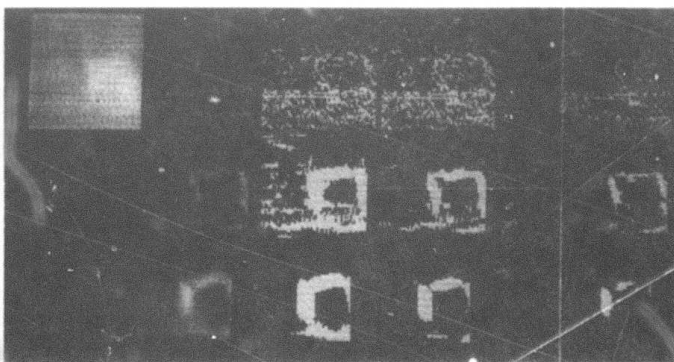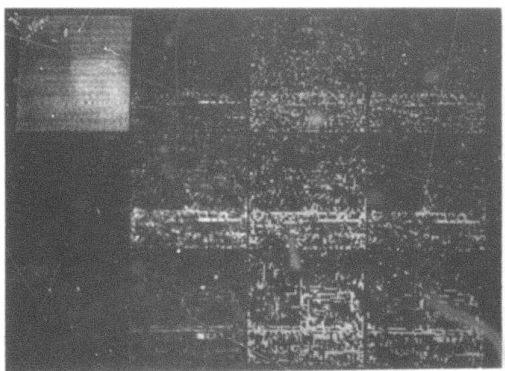
44A

Figure 21 (continued)
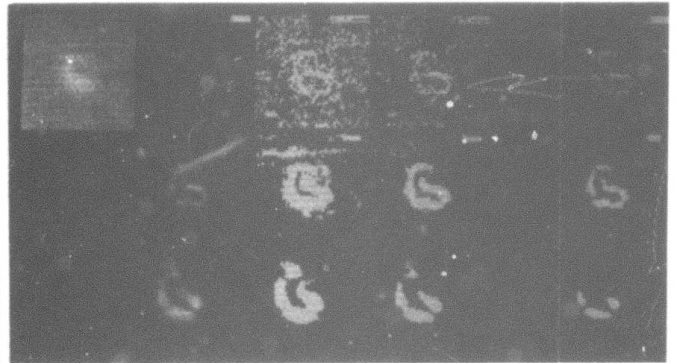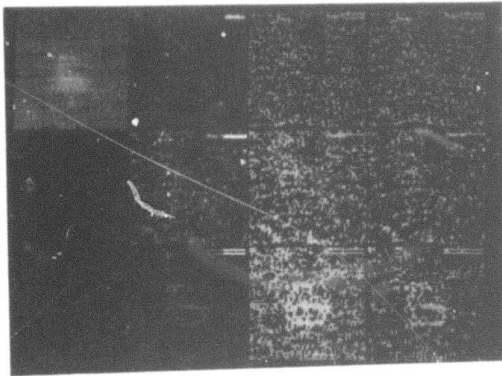
46A
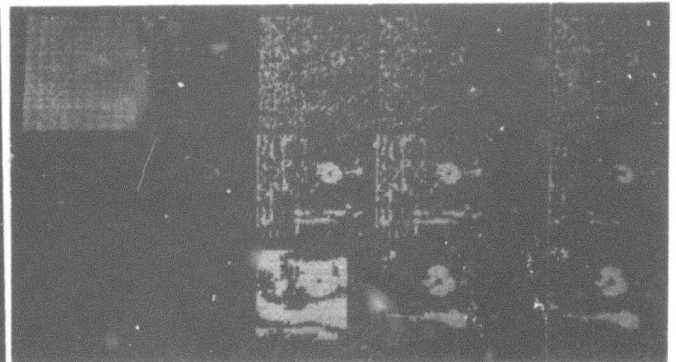
52A
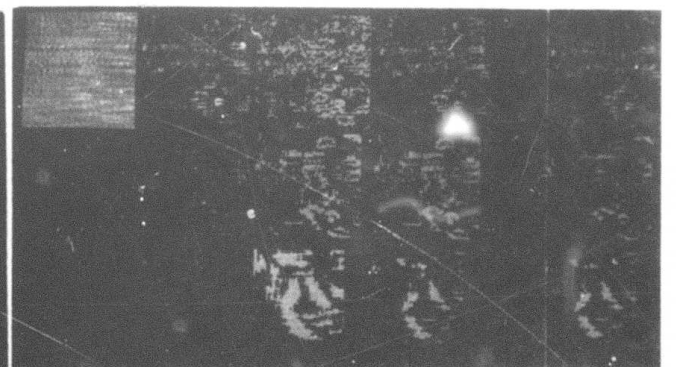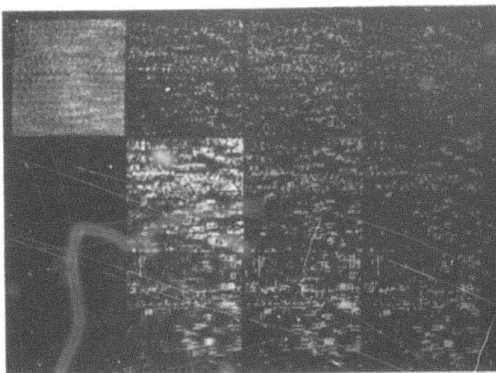
2N

Figure 21   (continued)
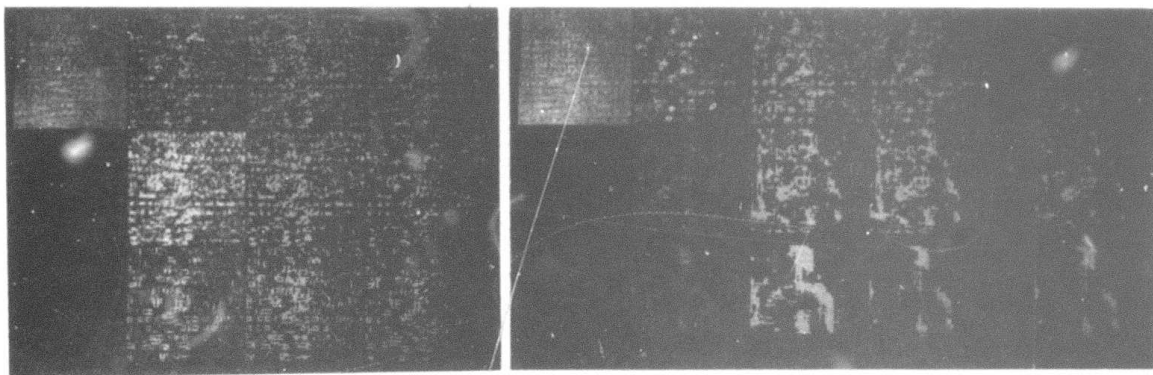
8N
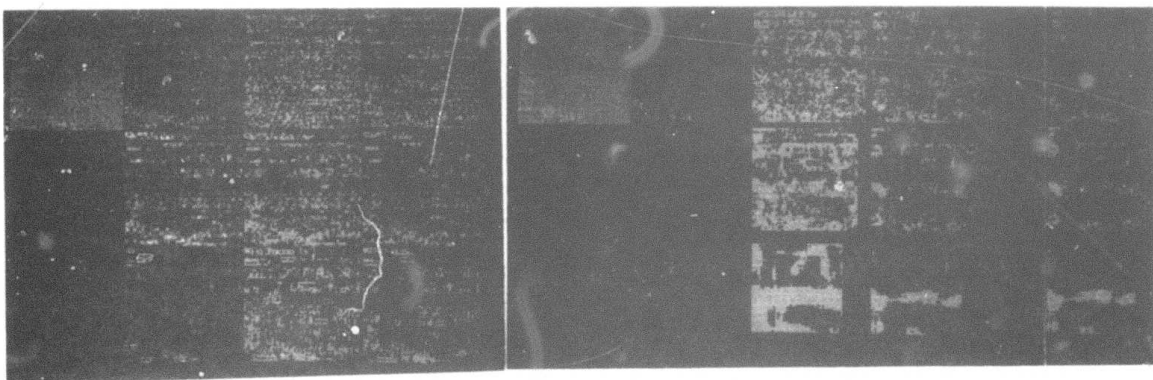
14N
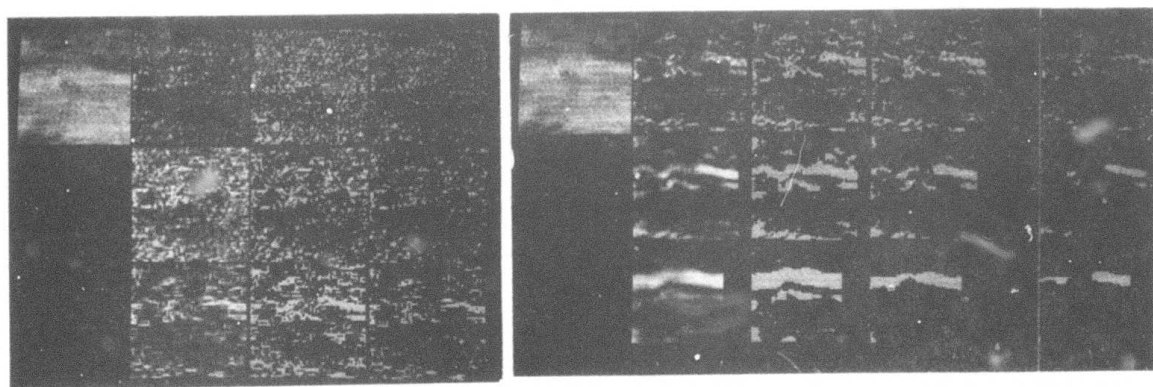
20N

Figure 21 (continued)
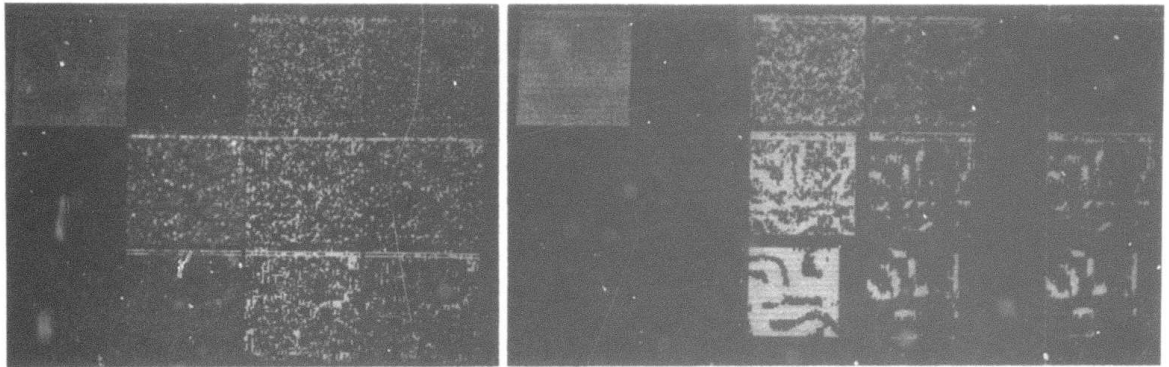
26N
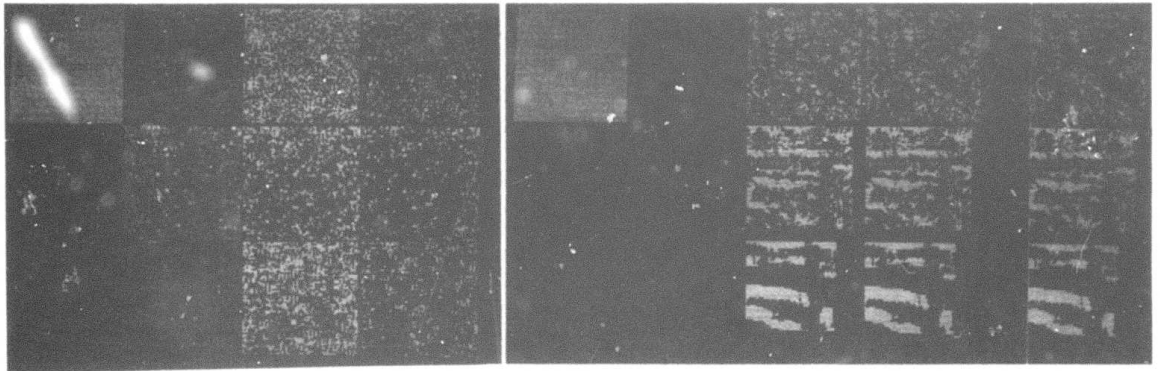
32N
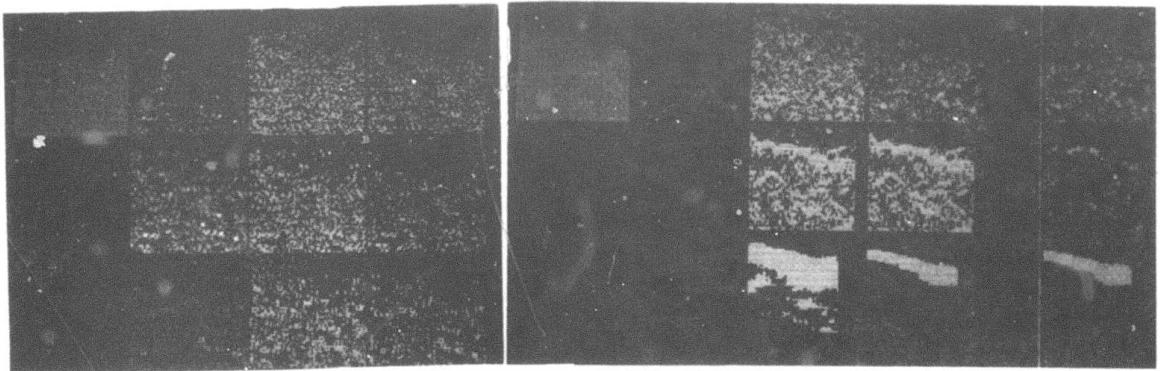
38N

Figure 21 (continued)
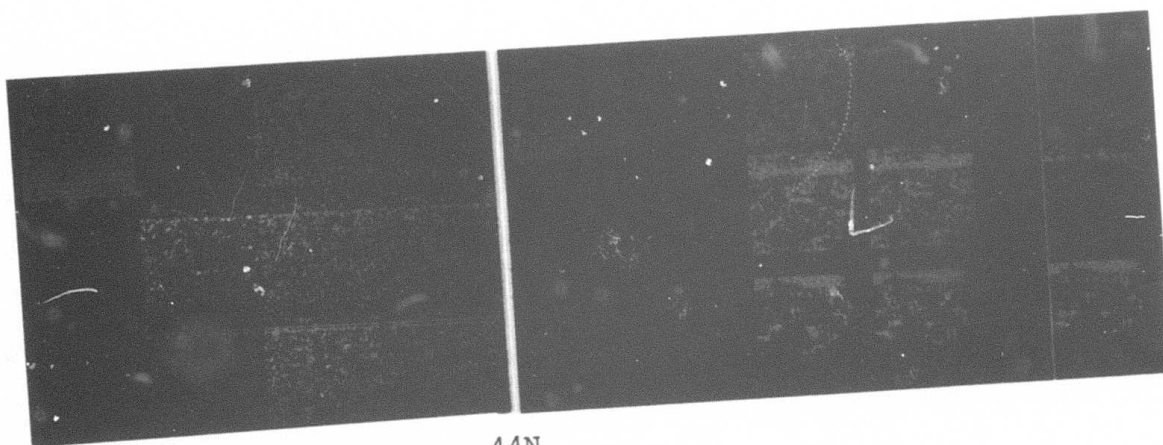
44N

Figure 21 (continued)

Laplacian: $|e - (a+b+c+d+f+g+h+i)/8|$, where the

neighborhood of e is

```
a b c
d e f t
g h i u
  v w
```

Roberts Gradient: $\max\{|a-e|, |b-d|\}$.

Three-by-three: $\max\{|a+b+c+-g-h-i|, |a+d+g-c-f-i|\}$

2x2 Difference:

$1/4*\max\{|d+e+g+h-f-t-i-u|, |b+c+e+f-h-i-v-w|\}$.
(In other words, the value corresponds to the
maximum of the differences between 2x2 aver-
ages over adjacent pairs of horizontal and
vertical neighborhoods.)

4x4 Difference: This is the same as the 2x2 diff-
erence except that averages are taken over 4x4
neighborhoods.

8x8 Difference: The same as the previous except
that averages are taken over 8x8 neighborhoods.

In order to select high edge values, three per-
centiles were chosen -- 80%, 90% and 95%. Figure 21 also
illustrates the masks consisting of points whose edge
values were in the top 20%, 10% and 5%, respectively. The
gray levels (in the original images) at these points were
histogrammed and the means and modes tabulated (Table 3).
The mean values were used as thresholds on the original
images. Figure 22 shows the resulting thresholded images.

Table 3. Gray level statistics for points of high
edge value based on 80th, 90th, and 95th
percentiles for five edge operators. ("OP"
codes represent in order: Laplacian,
Roberts gradient, 3x3 gradient, 4x4 averages
difference, 8x8 averages difference.
IRN refers to the image reference number in
the NVL data base.)

Table 3.

OP	PIC	%	MEAN	MODE	ST DEV	%	MEAN	MODE	ST DEV	IRN
LAP11	T3	80	25	29	6.38	90	26	29	10.05	3
LAP11	T5	80	23	25	8.62	90	24	2b	10.00	6
LAP11	T9	80	24	27	9.14	90	26	27	10.35	11
LAP11	T14	80	26	30	9.27	90	30	27	11.69	16
LAP11	T17	80	24	24	6.59	90	25	25	7.38	22
LAP11	T21	80	23	24	3.22	90	24	26	4.08	28
LAP11	T25	80	26	24	4.02	90	26	25	4.43	34
LAP11	T31	80	21	21	5.98	90	21	24	5.99	45
LAP11	T35	80	21	23	3.96	90	22	23	4.67	52
LAP11	T40	80	24	26	3.72	90	26	24	4.34	57
LAP11	R1	80	23	23	3.10	90	23	29	4.42	3
LAP11	R5	80	24	28	5.71	90	28	29	9.50	22
LAP11	R8	80	25	26	5.24	90	26	29	6.02	24
LAP11	R10	80	24	24	6.18	90	24	26	6.85	31
LAP11	R13	80	22	22	2.91	90	24	26	3.41	34
LAP11	R17	80	23	23	3.67	90	23	18	3.67	47
LAP11	R22	80	24	25	6.47	90	24	25	7.24	48
LAP11	R24	80	24	24	2.65	90	22	26	6.08	57
LAP11	A1	80	24	24	2.09	90	24	26	7.48	21
LAP11	A4	80	27	25	6.11	90	22	14	6.24	24
LAP11	A5	80	23	29	2.97	90	24	27	10.94	27
LAP11	A11	80	22	24	2.35	90	22	25	4.38	37
LAP11	A13	80	19	22	3.17	90	28	29	5.93	42
LAP11	A16	80	23	23	4.75	90	23	26	6.40	44
LAP11	A22	80	22	22	4.16	90	19	22	4.71	46
LAP11	A26	80	29	25	4.71	90	31	37	5.35	52
LAP11	N2	80	21	23	3.74	90	30	34	3.74	54
LAP11	N6	80	22	22	7.54	90	27	26	8.05	58
LAP11	N14	80	23	23	8.51	90	22	27	10.06	2
LAP11	N20	80				90	27	22	14.96	8
LAP11	N23	80	22	25	4.37	90	23	37	10.55	14
LAP11	N32	80	23	24	4.11	90	20	25	3.84	20
LAP11	N38	80	23	23	2.55	90	23	23	3.18	26
LAP11	N44	80	23	26	3.90	90	26	26	4.41	32
LAP11	N50	80	22	22	3.08	90	25	25	4.55	38
LAP11	N56	80	21	23	4.12	90	21	23	3.70	44

OP	PIC	%	MEAN	MODE	ST DEV	%	MEAN	MODE	ST DEV	IRN
ROB	T3	80	25	28	8.10	90	26	26	9.67	3
ROB	T5	80	22	19	7.99	90	25	17	10.61	6
ROB	T9	80	26	21	7.47	90	27	26	10.67	11
ROB	T14	80	23	22	7.56	90	30	28	11.66	16
ROB	T17	80	24	25	2.73	90	25	25	6.62	22
ROB	T21	80	23	25	2.71	90	24	23	2.73	28
ROB	T25	80	26	21	5.77	90	29	27	4.01	35
ROB	T31	80	20	20	3.39	90	27	21	5.94	45
ROB	T40	80	24	20	7.32	90	21	20	4.21	52
ROB	R1	80	23	20	4.99	90	25	26	3.70	57
ROB	R3	80	23	20	7.51	90	24	23	8.52	3
ROB	R6	80	27	26	5.98	90	23	26	5.77	6
ROB	R8	80	24	25	3.26	90	27	23	5.16	22
ROB	R13	80	23	21	7.00	90	24	24	6.52	24
ROB	R17	80	22	21	5.45	90	24	21	3.33	31
ROB	R22	80	23	21	2.59	90	22	29	3.60	34
ROB	R24	80	23	24	7.73	90	23	20	7.16	48
ROB	A1	80	24	24	7.98	90	23	23	5.86	55
ROB	A4	80	24	21	7.91	90	23	25	2.89	57
ROB	A5	80	21	22	4.63	90	21	21	6.91	21
ROB	A11	80	27	21	4.11	90	27	19	5.23	24
ROB	A14	80	22	22	3.35	90	23	23	9.14	27
ROB	A16	80	20	22	5.05	90	23	25	4.27	42
ROB	A22	80	28	23	2.55	90	22	21	5.34	44
ROB	A26	80	21	25	7.70	90	22	21	5.49	46
ROB	N2	80	20	21	2.87	90	17	21	6.53	52
ROB	N6	80	26	21	3.62	90	29	26	3.82	54
ROB	N14	80	21	18	3.08	90	21	29	3.83	58
ROB	N20	80	21	21	2.46	90	22	17	7.25	2
ROB	N26	80	22	21	3.00	90	21	21	4.56	8
ROB	N32	80	19	18	3.41	90	26	21	9.63	14
ROB	N44	80	22	21		90	21	22	4.04	20
ROB	N50	80	21	21		90	19	17	3.89	26
ROB	OCN	0	20	21		90	22	23	3.76	32
						90	21	21	3.31	38
						90	20	21	4.02	44

OP	PIC	%	MEAN	MODE	ST DEV	%	MEAN	MODE	ST DEV	IRN
THRE	T2	90	25	26	6.77	90	26	26	8.96	3
THRE	T3	90	23	21	8.31	90	25	26	9.80	6
THRE	T14	90	26	22	4.24	90	28	26	10.18	11
THRE	T17	90	24	23	7.95	90	31	23	11.40	16
THRE	T21	90	23	23	3.75	90	25	21	16.35	22
THRE	T25	90	27	24	3.57	90	24	23	3.03	28
THRE	T35	90	20	19	8.24	90	23	24	5.11	34
THRE	T40	90	24	21	6.95	90	28	29	5.45	45
THRE	K1	90	23	22	8.63	90	21	20	3.86	52
THRE	K5	90	27	25	7.23	90	25	25	8.19	57
THRE	K6	90	25	25	4.50	90	24	21	5.13	3
THRE	K13	90	24	20	7.04	90	23	25	4.69	6
THRE	K17	90	26	21	6.48	90	26	25	6.83	22
THRE	K24	90	22	21	8.89	90	24	21	3.98	24
THRE	A1	90	24	22	9.09	90	27	26	6.60	34
THRE	A4	90	24	21	6.72	90	22	25	2.79	47
THRE	A5	90	23	21	7.44	90	22	21	2.97	48
THRE	A13	90	25	21	4.03	90	24	17	5.24	55
THRE	A14	90	24	21	5.53	90	23	22	7.92	57
THRE	A20	90	22	20	8.05	90	25	21	4.34	21
THRE	A22	90	24	21	2.24	90	24	21	5.61	24
THRE	N2	90	23	21	4.60	90	22	23	6.19	27
THRE	N6	90	27	28	10.11	90	29	27	3.25	32
THRE	N14	90	21	21	4.19	90	21	28	6.72	42
THRE	N26	90	19	18	2.31	90	27	32	8.57	46
THRE	N32	90	22	21	3.45	90	22	21	8.30	52
THRE	N44	90	21	21	3.82	90	19	18	4.71	54
THRE	N50	90	21	21	5.21	90	21	21	3.33	58
THRE	N56	90	20	20		90	19	16	2.54	8

GP	PIC	%	MEAN	MODE	ST DEV	%	MEAN	MODE	ST DEV	%	MEAN	MODE	ST DEV	IRN
DIF	T3	80				90	29	30	10.14	95	29	26	7.20	3
DIF	T5	80				90	28	23	9.31	95	29	23	7.41	6
DIF	T9	80				90	27	23	4.73	95	30	23	10.37	11
DIF	T14	80				90	25	26	3.12	95	30	26	10.05	16
DIF	T17	80				90	23	23	3.78	95	22	23	4.55	28
DIF	T21	80				90	21	23	4.07	95	23	24	4.24	34
DIF	T25	80				90	19	30	3.22	95	23	31	5.52	45
DIF	T31	80				90	13	18	3.66	95	21	18	3.22	52
DIF	T35	80				90	24	22	4.64	95	24	26	3.68	57
DIF	T40	80				90	29	24	4.48	95	28	24	3.22	36
DIF	R1	80				90	25	28	5.23	95	22	28	5.08	24
DIF	R3	80				90	24	25	2.67	95	24	23	3.16	31
DIF	R6	80				90	22	24	2.68	95	18	24	4.68	34
DIF	R10	80				90	23	29	2.29	95	23	18	4.76	47
DIF	R13	80				90	22	21	2.18	95	23	23	2.87	48
DIF	R17	80				90	23	21	2.19	95	23	23	4.50	55
DIF	R22	80				90	25	24	3.07	95	25	25	5.18	57
DIF	R24	80				90	23	26	5.70	95	23	26	4.95	21
DIF	A1	80				90	27	36	4.88	95	30	37	3.22	24
DIF	A4	80				90	22	21	4.72	95	32	36	5.07	27
DIF	A5	80				90	23	23	4.34	95	34	21	3.03	42
DIF	A11	80				90	29	27	4.18	95	33	23	4.53	44
DIF	A13	80				90	30	26	3.36	95	23	25	5.32	46
DIF	A14	80				90	21	20	3.77	95	30	30	5.57	54
DIF	A16	80				90	29	32	3.04	95	23	31	6.30	58
DIF	A22	80				90	19	21	6.01	95	30	29	6.36	32
DIF	A26	80				90	30	18	2.53	95	22	32	1.60	8
DIF	N2	80				90	29	23	1.83	95	21	22	4.53	40
DIF	N6	80				90	20	21	2.82	95	29	20	1.83	26
DIF	N14	80				90	22	20	2.59	95	22	21	3.50	32
DIF	N50	80				90	19	20	3.19	95	17	17	2.91	56

GP PIC % MEAN MODE ST DEV % MEAN MODE ST DEV % MEAN MODE ST DEV % MEAN MODE ST DEV IRN

Figure 3.7 Thresholded template matrix as shown by the mean gray level of the points in the ...

Figure 22.   Thresholded images, using as threshold
the mean gray level of the points in the
80th edge value percentile, for the
following edge detectors:

Laplacian   2x2
                   difference

Roberts   4x4
 gradient   difference

3x3   8x8
 gradient   difference
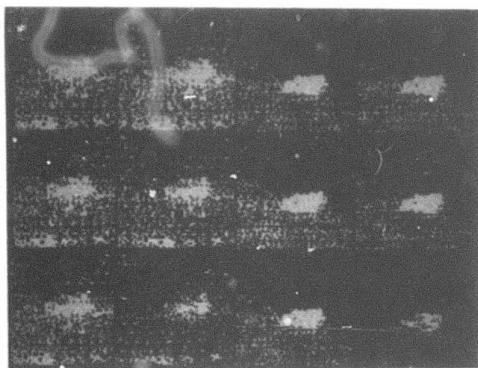
3T   6T   11T   16T
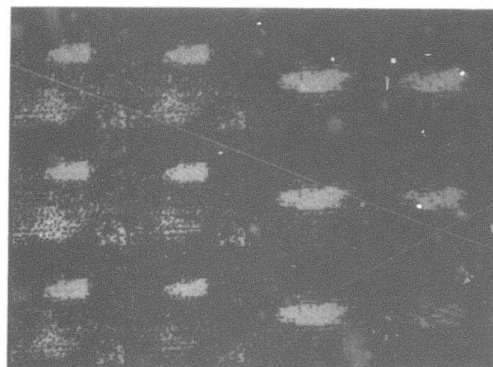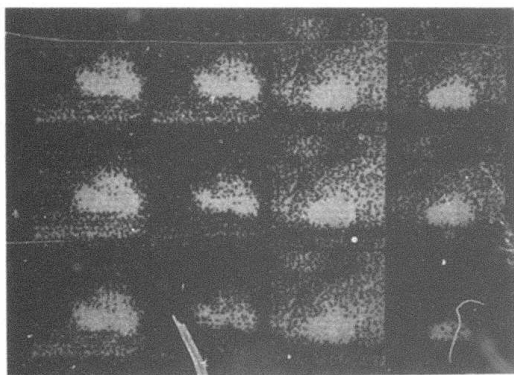
22T   28T   34T   45T
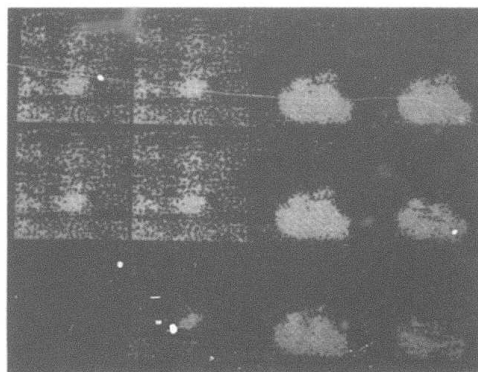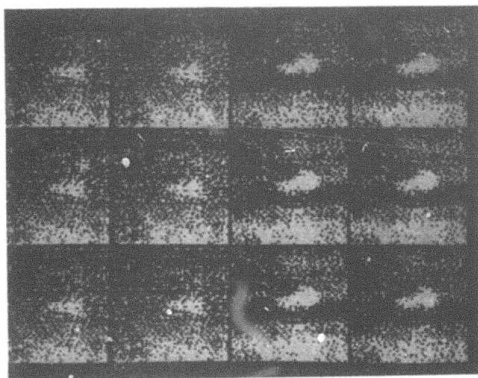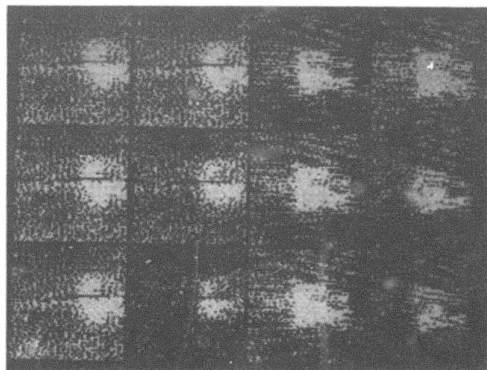
3R   6R   22R   24R

Figure 22 (continued)

31R   34R   47R   48R

21A   24A   27A   37A
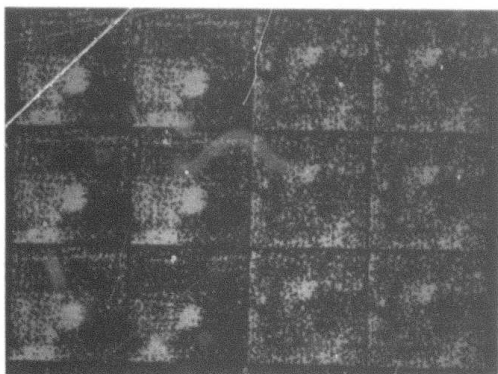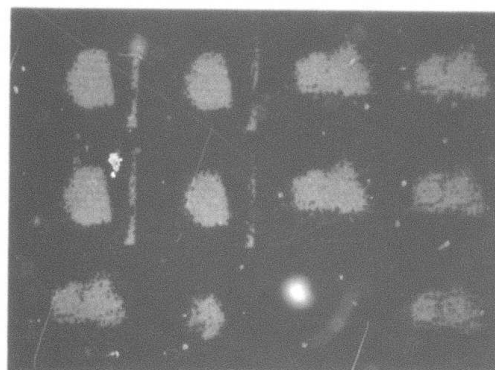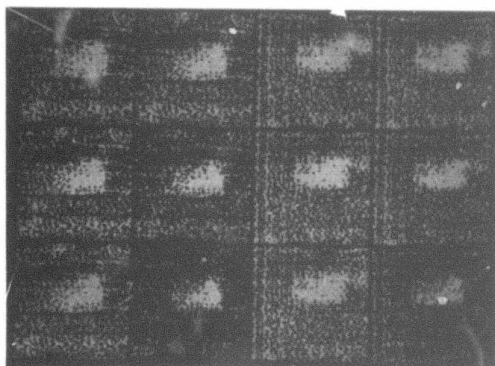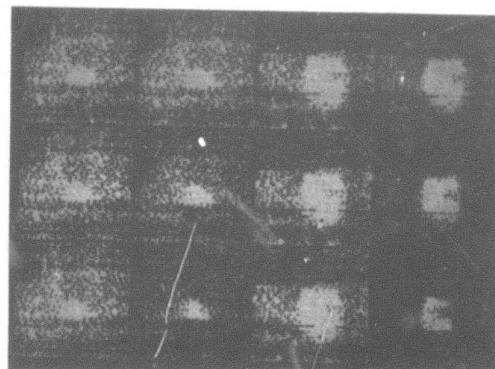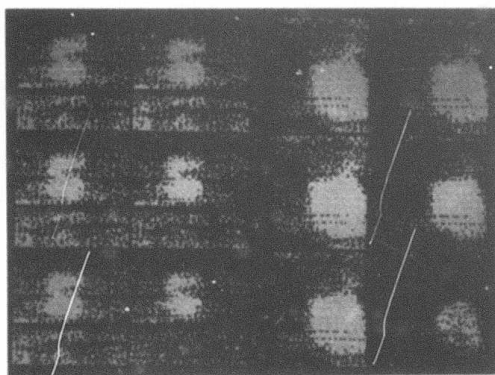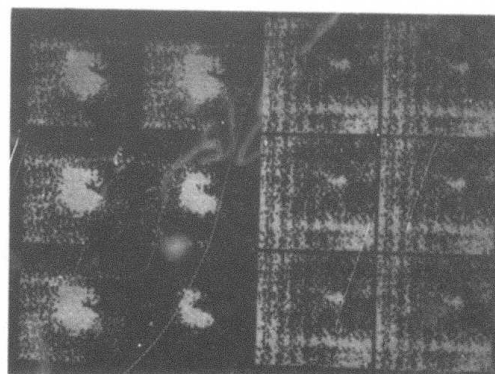
42A   44A   46A   52A

Figure 22 (continued)
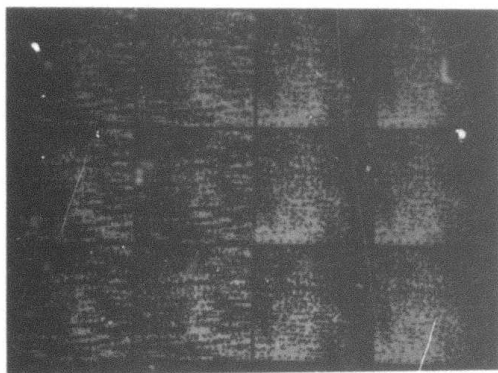
2N   8N
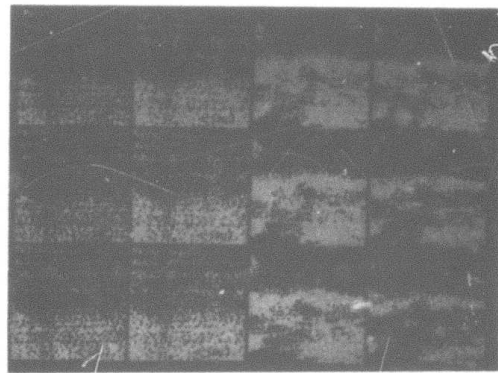
14N   20N
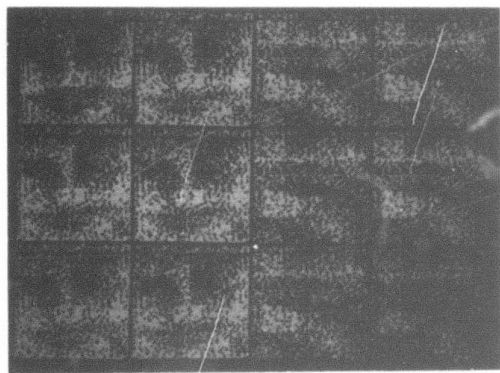
26N   32N
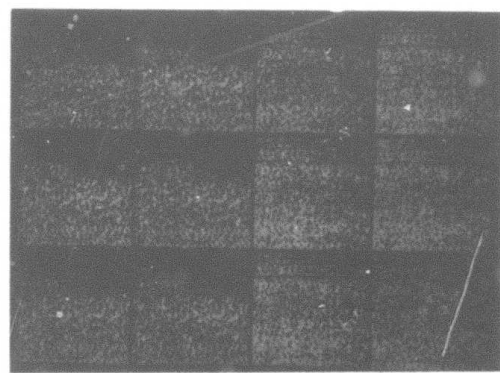
38N   44N

Figure 22 (continued)

Discussion

A number of parameters were introduced in this study, including the choice of edge operator, and the percentile cutoff. The Laplacian gave poor results due to the "rampiness" of the edges. The gradients produced reasonable edge masks but predicted different thresholds (i.e., different means of high edge value gray levels). The 4x4 operator produced good thresholds overall. This may be explained by noting that the ramp width as determined from the Roberts gradient was approximately 3. Thus operators based on 3x3 neighborhoods or smaller could not span the ramp and give accurate gradient values. The 4x4 difference, on the other hand, does span the ramp edge more effectively. If the neighborhood is too big, as in the 8x8 difference, the edge operator will be biased in favor of the background (since the object shape is usually convex) and will on occasion span the whole object (thus decreasing its response at the actual edge).

The choice of a percentile cutoff for the edge values is much more difficult to assess. Clearly, if an image contains no object then no percentile should be chosen since the operator is responding only to noise. The use of a cutoff assumes that the highest edge values will correspond to object/background edges rather than to noise. This is valid in the case of the difference of averages operators, since these operators will not respond as well to local noise as they do to object edges. The cutoff should therefore be at the edge value likely to separate out almost all of the object edge values. Clearly, the optimal value depends on

target size. However, in practice, the 80th percentile of the 4x4 operator produced a reliable sample which contained somewhat more background edge points than object points, resulting in a low threshold; but this was deemed acceptable because of the subsequent noise cleaning process which tended to smooth tattered object boundaries.

Overall, this automatic thresholding technique provided reasonable thresholds and produced good segmentations for later processing. Computationally, the procedure involved three passes over the input image and one pass over the intermediate edge image. During the first image pass, the edge operator is applied and an edge image created. Simultaneously, an edge value histogram is computed. The 80th percentile value is then compared. During the second pass, both the input image and the edge image are read. The gray level values of those edge points whose edge values are at or above the 80% cutoff value are histogrammed. The mean of the histogrammed points serves as the threshold for the third pass. In a dynamic environment, producing thirty images per second, it should be possible to apply the three different passes in pipeline fashion to three consecutive images (assuming that the gray level statistics remain stable over the period necessary to process three images). The storage requirements would be reduced to the number of lines necessary to compute the operator. This dynamic approach will be tested with a real-time sequence of images during the next quarter.

An alternative computational approach determines the threshold in a single pass over the input image, at the cost of storing a 2-D histogram within an array of counters. If the gray level value at the current image point is i and the edge operator is j, then the (i,j)th counter is incremented. Now, high edge values correspond to high-index rows in the 2-D histogram array. The row sums form the edge value histogram, whose 80th percentile is then chosen. Next, the column sums are formed for all rows at or above the 80% cutoff. These sums constitute the gray level histogram for high edge values. The mean of this histogram is the desired threshold.

E1. Edge Reinforcement Prior to Noise Cleaning

Noise cleaning operations, in particular, parallel shrink/expand algorithms, will delete points from ragged edges of objects. The result is that an object will be displayed with fewer points than were proposed by the thresholding step. One way of avoiding this is to choose a slightly more generous threshold, thus adding in more background points which presumably are later rejected during noise cleaning. Unfortunately, this strategy adds in noise points all over the image. A technique which adds only points at or near the boundary of objects is preferable. Such points generally have high gradient value. Section B suggested the following technique: compute the threshold automatically as described in Section D, and use a combined (gray level, gradient value) threshold to include high gradient value points which don't quite exceed the gray level threshold value. In practice, this may be implemented using the 2-D (gray level, gradient) histogram; such histograms are shown in Figure 23. On such a histogram, a vertical line corresponds to a gray level threshold, while an oblique line corresponds to a combined (gray level, gradient value) threshold. Image points whose (gray level, gradient value) pair lies to the left of BC are considered to be above threshold. Figure 24 illustrates this for several values of θ. Note that the effect of varying θ is mainly the accretion or loss of edge points. Clearly, the implementation of a combined threshold can be accomplished in the single

Figure 23.   Two-dimensional (gray level, gradient)
             histograms. The displayed histogram
             value at row R and column C is the (log
             scaled) number of image points which
             have edge value R and gray level C.

Key: H_x denotes the two-dimensional histogram for
      edge detector x.

Edge detector	Denoted by subscript
Laplacian	L
Roberts gradient	R
3x3 gradient	3
2x2 difference	2
4x4 difference	4
8x8 difference	8

Original
  image

L	H_L	2	H_2
R	H_R	4	H_4
3	H_3	8	H_8

Chromosome

11T

22T

6R

Figure 23 (continued)

47R

48R

21A

27A
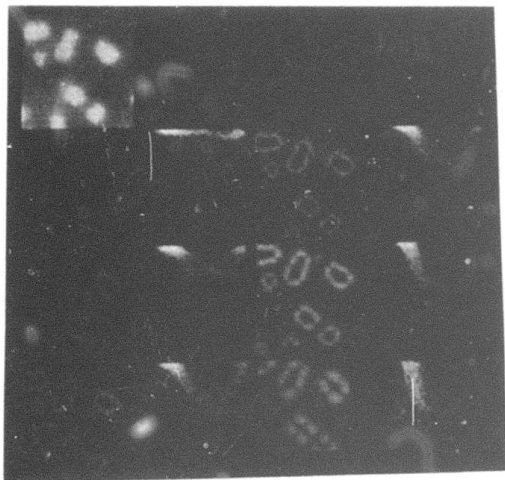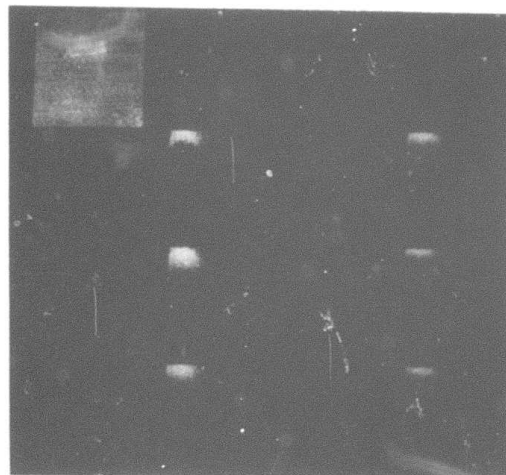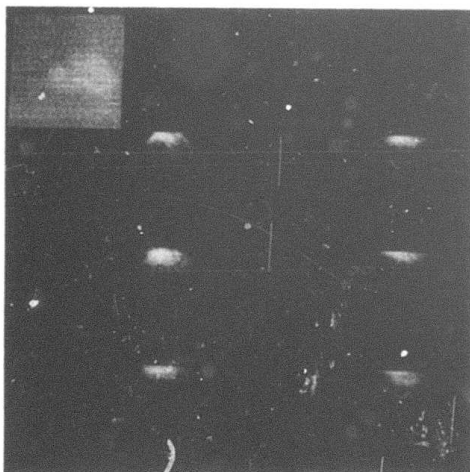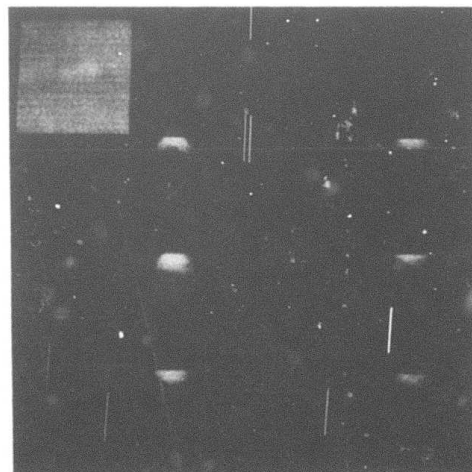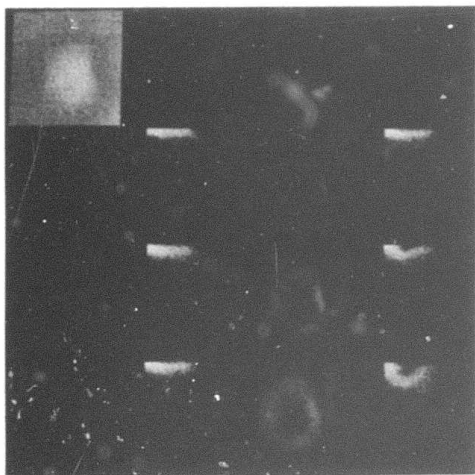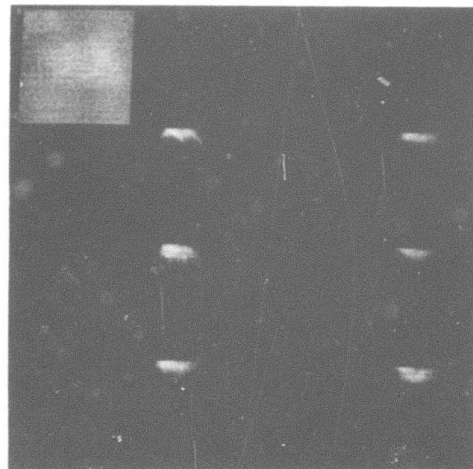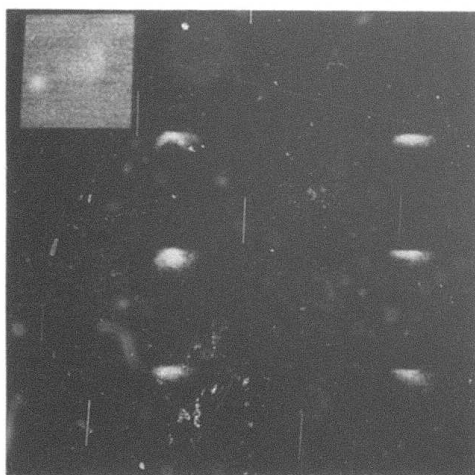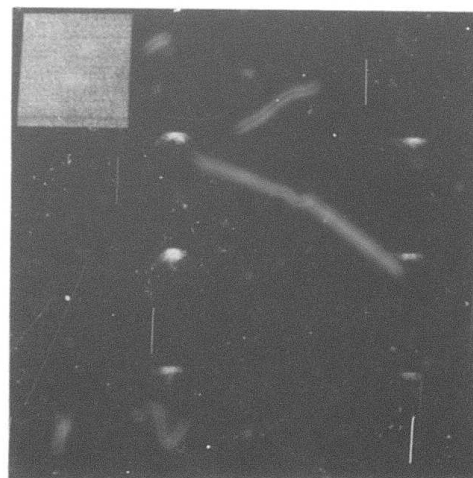
Figure 23 (continued)
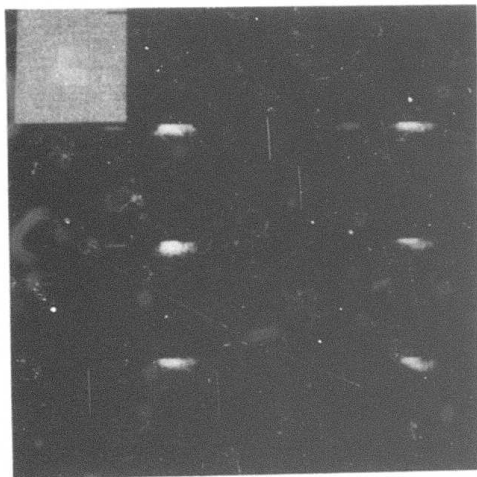
46A
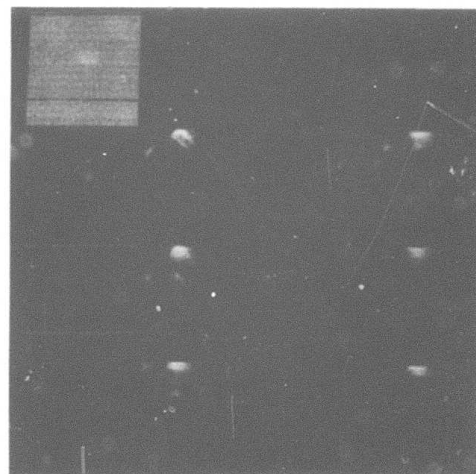
58A

26N

50N
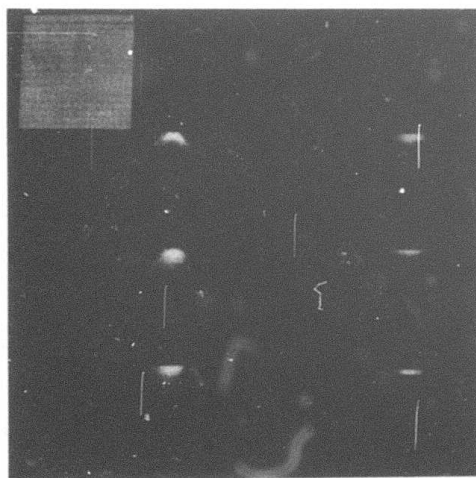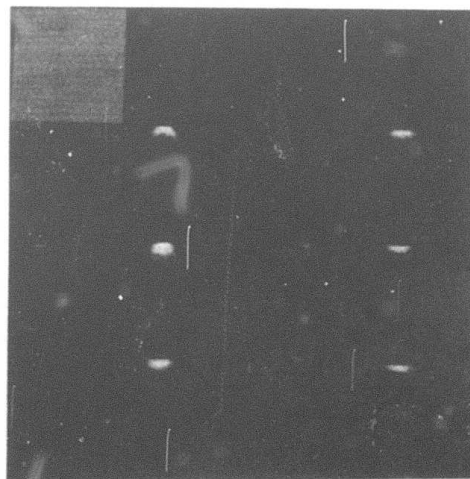
Figure 23 (continued)

Figure 24. Thresholding using combined (gray level gradient) thresholds for the following values of θ.

-30°	10°
-20°	20°
-10°	30°
0°	
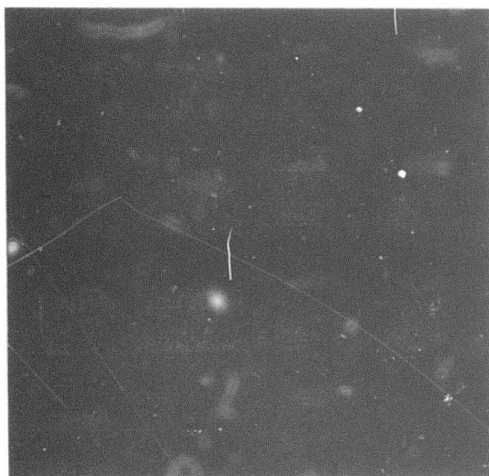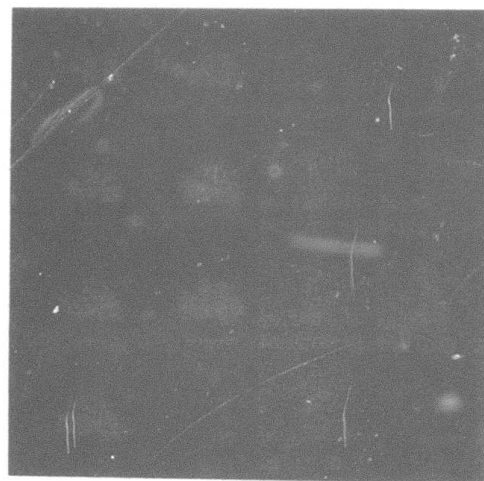
3T   11T   22T   34T
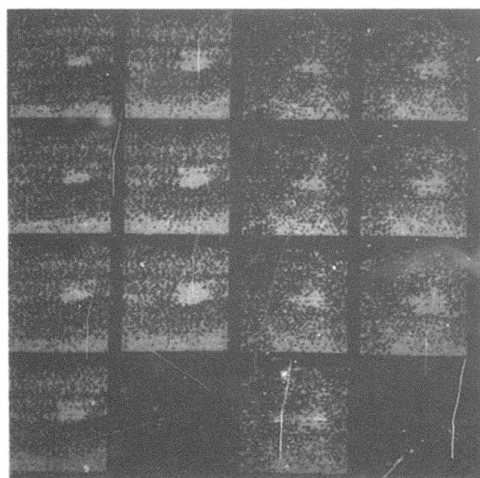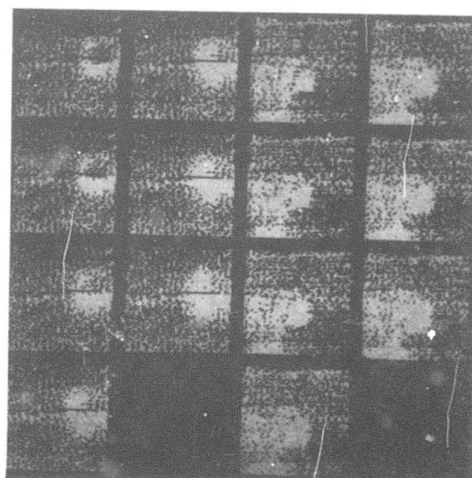
52T   3R   22R   31R

Figure 24 (continued)
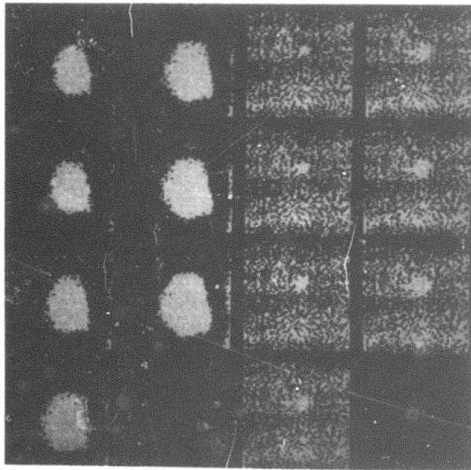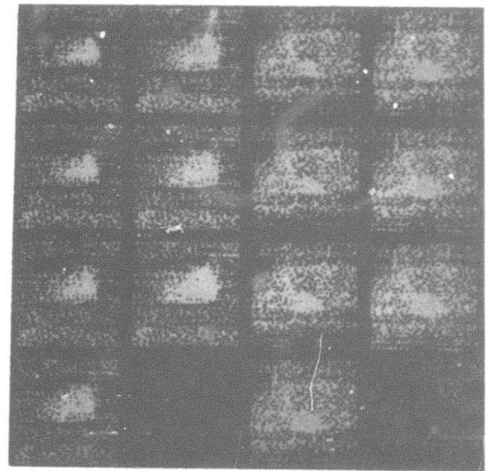
47R   55R   21A   27A
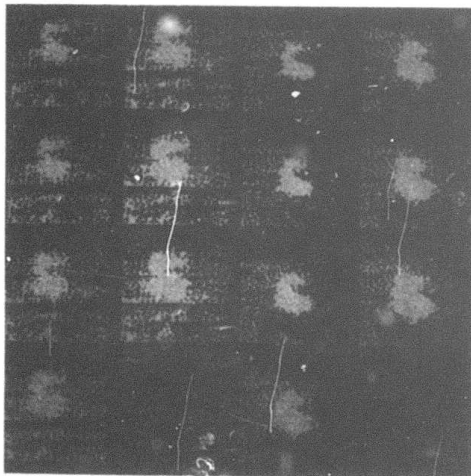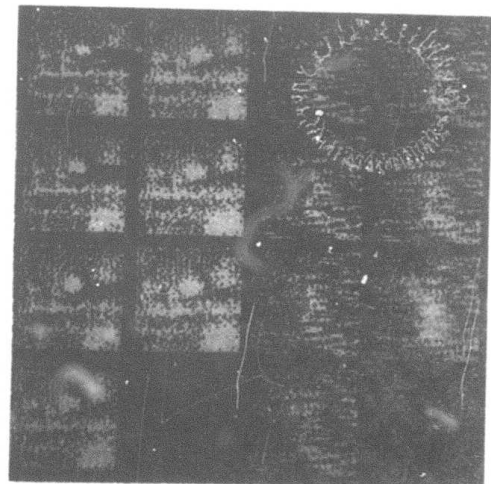
42A   46A   54A   2N

Figure 24 (continued)
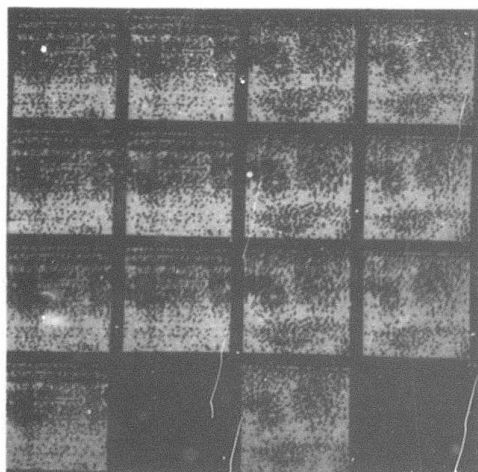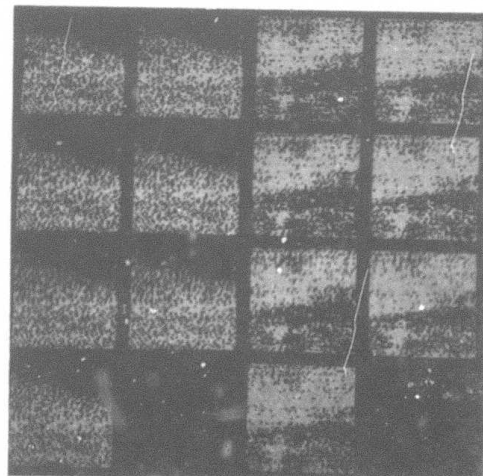
14N   26N   38N   50N

Figure 24 (continued)

thresholding pass. Further study of the model is needed
to predict the appropriate value of θ for this type of
thresholding procedure.

E2. Noise Cleaning by Averaging

The images in the data set are oversampled at a ratio of over 2:1 for the purpose of scaling the horizontal and vertical axes. The processed windows, however, were sampled down 2 to 1. The resulting windows exhibit moderate to severe high frequency noise. An effort was made to reduce this noise by producing windows based on 2x2 averaging rather than sampling. Thus, instead of discarding every other row and column, each pixel in the sampled image was the average over a (disjoint) 2x2 neighborhood in the original image. The results (Figure 25) show that a smooth, less noisy image was produced and that row dropouts were partially eliminated. However, the images seemed to have less contrast. Further experiments will determine if averaged windows should replace the sampled windows.
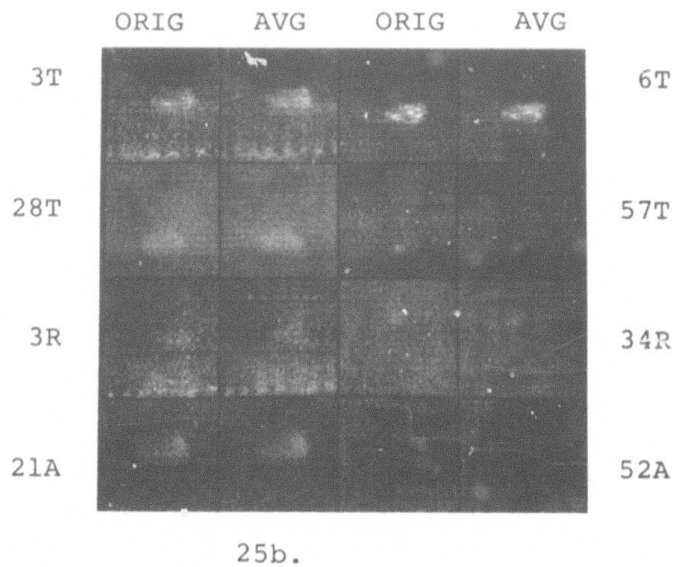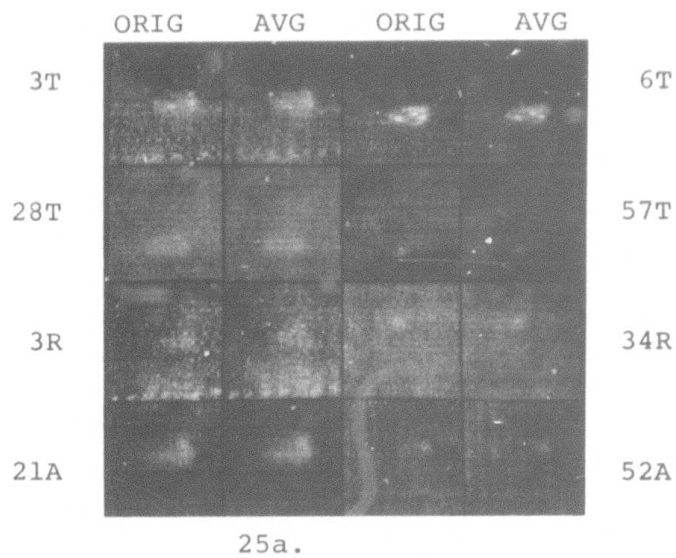
Figure 25.   Sampling vs. averaging in target windows.
             a.   The computed average using truncation.
             b.   Same, using rounding.

E3. Noise Region Filtering by Simultaneous Local Operations

The result of thresholding a new (unsmoothed) image is a binary valued image with object points identified (nominally) by the value 1 and background points by the value 0. In general, one often obtains isolated points or small regions corresponding to spurious identifications in both the background and object regions (see Figure 22). Such regions, which will be called noise regions, are recognizable by their small size and their isolation, especially since it has been postulated that object regions are compact. Note that these noise regions are artifacts of the thresholding and may not be readily visible in the original image.

One method of eliminating noise regions is to preprocess the image by smoothing. Preprocessing algorithms based on blurring and median filtering will be studied more extensively in the next quarter. Another approach is to postprocess the thresholded image to delete the noise regions. This section discusses postprocessing techniques which eliminate small and/or non-compact regions from a thresholded image.

The method which was investigated consists of multiple applications of two processes -- "shrink" and "expand" -- for example, two shrinks followed by two expands. The purpose of the sequence of shrinks is to shrink objects in a uniform manner so that small or insubstantial objects disappear entirely. The sequence of expands is meant to

regrow the remaining shrunken objects to their original
size. The result of the shrinks/expands is the elimination
of small regions (presumed to be noise regions).

Each shrink or expand requires the simultaneous
or "parallel" application of a local replacement rule at
every point of the thresholded image. This means that all
transform decisions are made on untransformed data, as dis-
tinguished from the sequential application of the trans-
formation rule in a raster fashion with transformed point
values replacing the original values as they are computed.

The form of the shrink rule is as follows: Rewrite
each 1 as 0 if any (at least one) of its neighbors is
already 0. Zero values are unchanged. (The 4-neighbor
case treats, as neighbors, points horizontally or vertically
adjacent to the given point; the 8-neighbor case includes
the points diagonally adjacent as well.) Such a rule de-
creases the number of 1's in the thresholded image; thus,
the image "shrinks". The rule can be interpreted as
eliminating all 1's adjacent to 0's. In fact, only 1's
surrounded by 1's will survive a shrink. Two shrinks
applied in succession will eliminate all 1's at a distance
of two or fewer raster units (city block or chessboard
distance) from the nearest 0. The number of successive
shrinks determines the minimum diameter of a region for it
to survive; e.g., one shrink eliminates all objects with
diameter two or less; two shrinks eliminate objects with
diameters of four or less.

The expand rule is similar to the shrink rule: rewrite a 0 as a 1 if any of its neighbors are 1's, but leave 1's unchanged. Thus points adjacent to 1's become 1's, thereby increasing the number of 1's. If we wish to restore objects (that are not eliminated) to about their original sizes, t shrinks should be followed by t expands. Such a shrink/expand sequence produces an image whose 1's correspond to (a subset of the) 1's in the untransformed binary image. Thus, for example, isolated 1's are eliminated, and objects joined by narrow necks of 1's may become disconnected. Also, thin protrusions from a region of 1's will disappear. Figure 26 illustrates the shrink/ expand algorithm for both the 4 and 8 neighbor cases and t = 1, 2, 3 (the numbers of shrinks and expands used). Figure 27 shows the effect that the choice of edge operator in threshold selection has on the subsequent noise cleaning.

The shrink rule as formulated was unsatisfactory because it tended to delete too much; it tended to produce regions all of the same shape (diamond-shaped); and it did not fill in pinholes. A generalization of the shrink rule was formulated as follows: delete a 1 if at least k of its neighbors are 0's (zeros remain unchanged). The original shrink rule corresponds now to k = 1. The generalized shrink is more conservative in that if k > 1, it takes more zero evidence to convert a 1 to 0. The generalized expand is analogously defined: Rewrite a 0 as 1 if it has at least k 1's as neighbors (ones remain unchanged). Note

Figure 26. Effects of iterating SHRINK/EXPANDS (S/E's).

    a. Original images - each column is a single image thresholded at four different values.

    b. 4-neighbor rule - one S/E

    c. 4-neighbor rule - two S/E's

    d. 4-neighbor rule - three S/E's

    e. 8-neighbor rule - one S/E

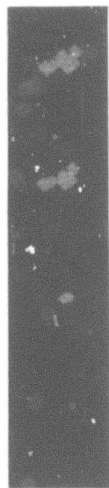   f. 8-neighbor rule - two S/E's

    g. 8-neighbor rule - three S/E's
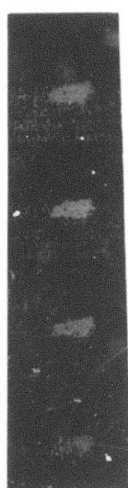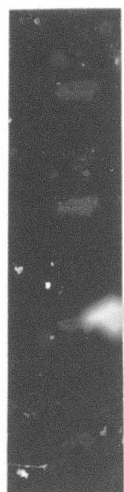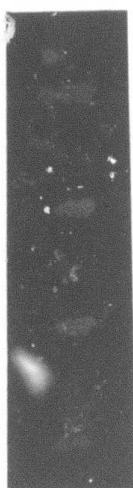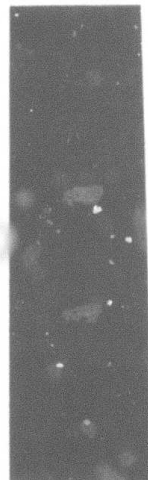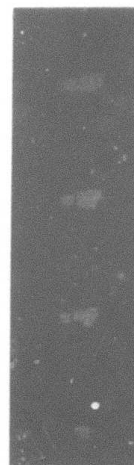
6T   a.   b.   c.   d.   e.   f.   g.
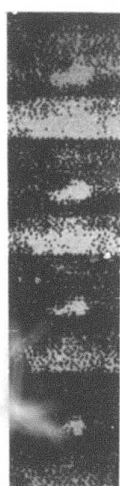
6R   a.   b.   c.   d.   e.   f.   g.
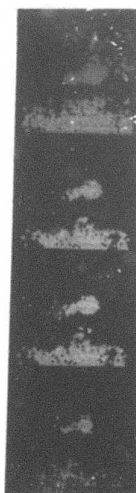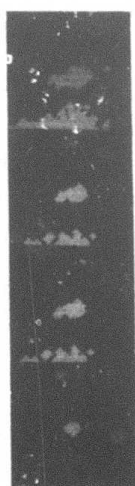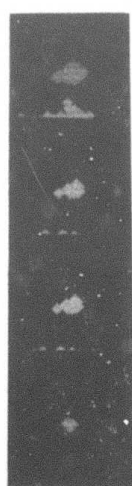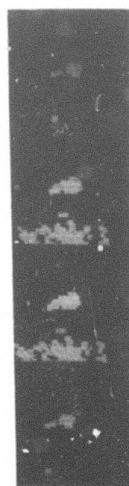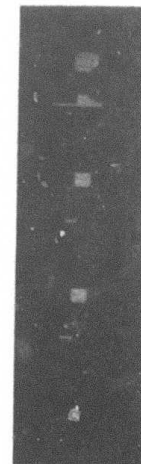
Figure 26 (continued)
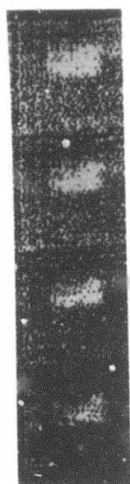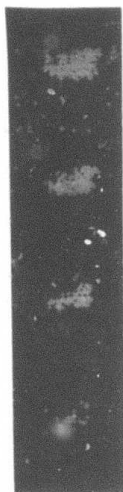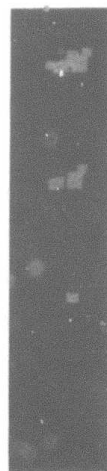
24A a.   b.   c.   d.   e.   f.   g.
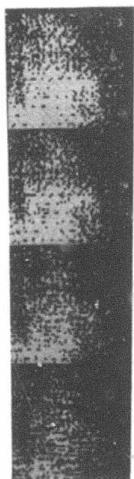
8N   a.   b.   c.   d.   e.   f.   g.
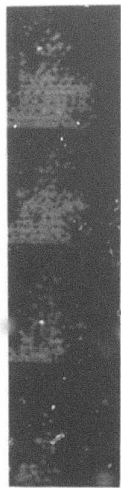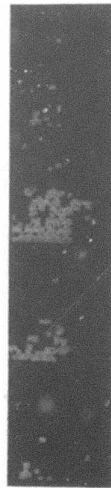
Figure 26 (continued)

Figure 27. SHRINK/EXPAND: comparison of edge detection
            operators (=1, t=2, 4-neighbor rule).

Key:

        Laplacian   2x2 difference

        Roberts gradient   4x4 difference

        3x3 gradient   8x8 difference
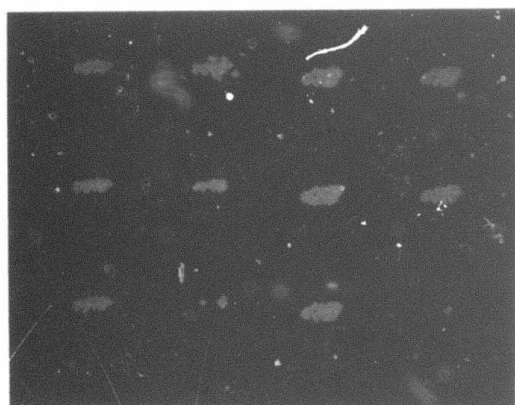
3T   6T   11T   16T
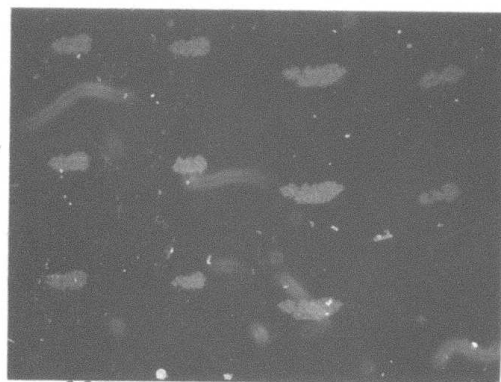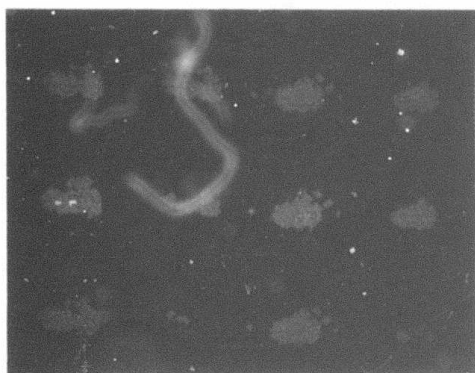
22T   28T   34T   45T
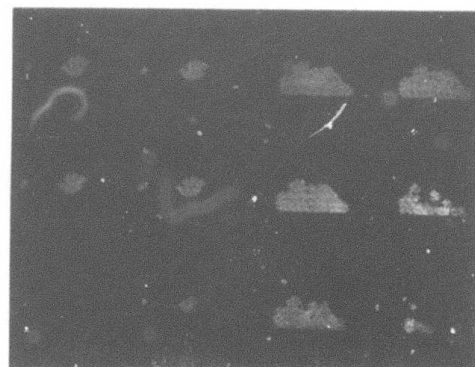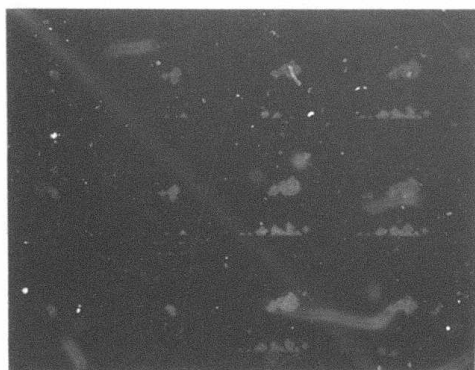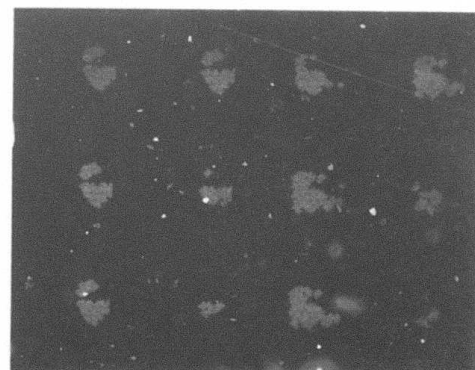
3R   6R   22R   24R

Figure 27 (continued)

31R   34R   47R   48R

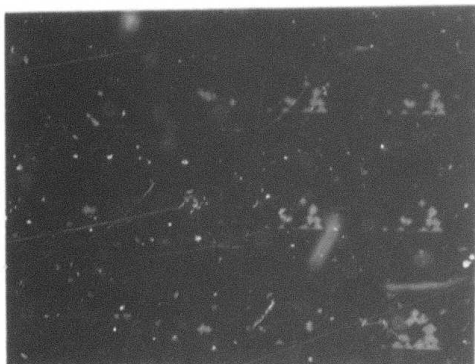21A   24A   27A   37A
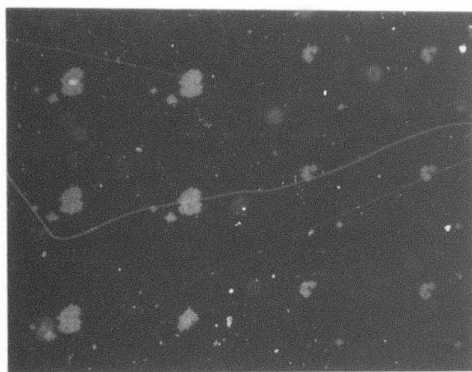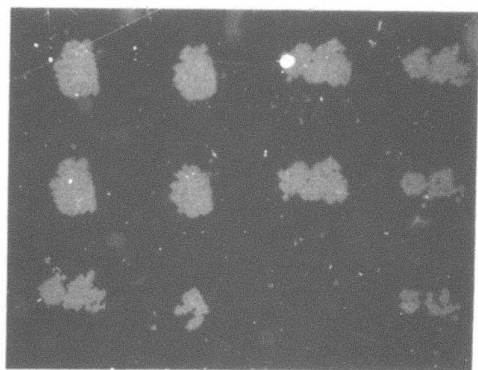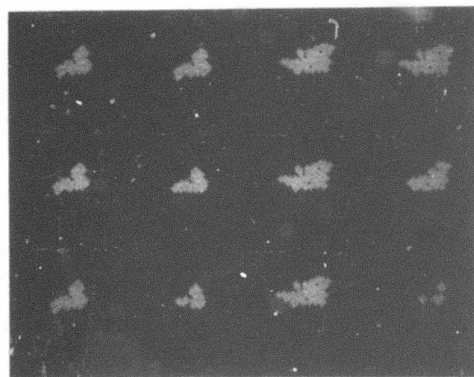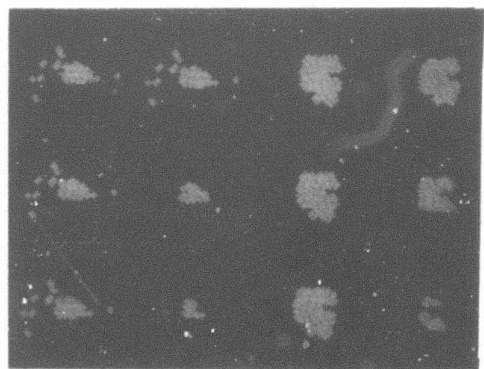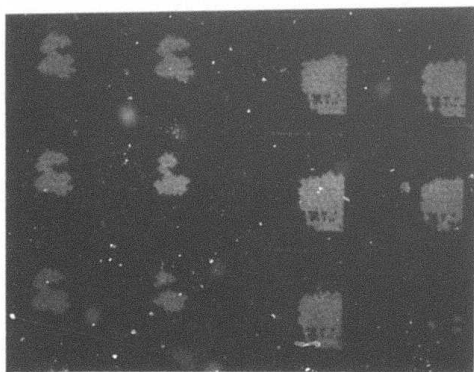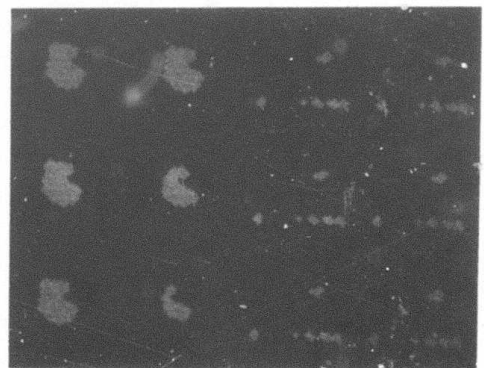
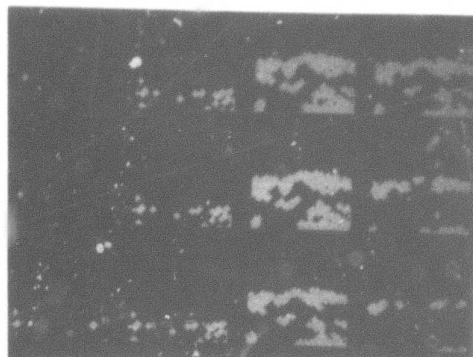42A   44A   46A   52A

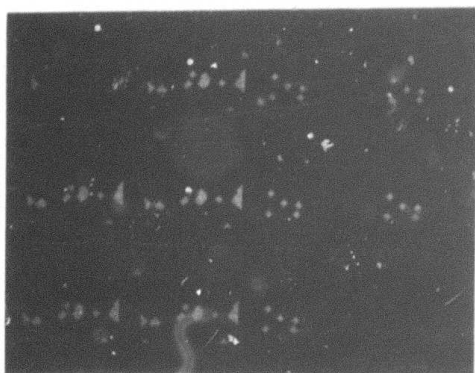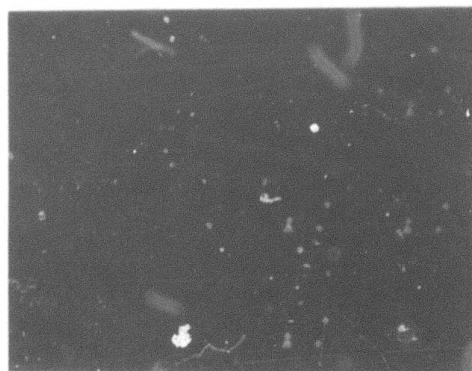Figure 27 (continued)

2N   8N

14N   20N

26N   32N

38N   44N

Figure 27 (continued)

that for increased k, the generalized expand rule is not quite as generous in providing new 1 values. However, it does fill pinholes in sufficiently large regions. Figure 28 provides a comparison for t = 1, 2 and k = 1, 2, 3. Figure 29 presents a further comparison based on the 4x4 edge operator used in threshold selection. It appears from these examples and from Figure 26 that the shrink/expand rule with t = 2 and k = 3 applied to each image point and its 8-neighbors provides efficient noise cleaning with most noise regions eliminated, pinholes filled, and only a modest amount of target shape distortion.

Figure 28. Leniency in SHRINK/EXPAND definitions
for windows thresholded by two methods.

    a. 4-neighbor rule, one S/E, k=1,2,3

    b. 8-neighbor rule, one S/E, k=1,2,3

    c. 4-neighbor rule, two S/E's, k=1,2,3

    d. 8-neighbor rule, two S/E's, k=1,2,3

ORIG   k=1   k=2   k=3

4x4
DIFF

3T

8x8
DIFF

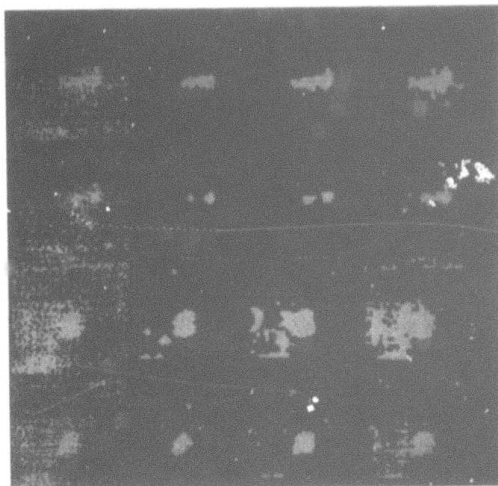4x4
DIFT

31R

8x8
DIFF
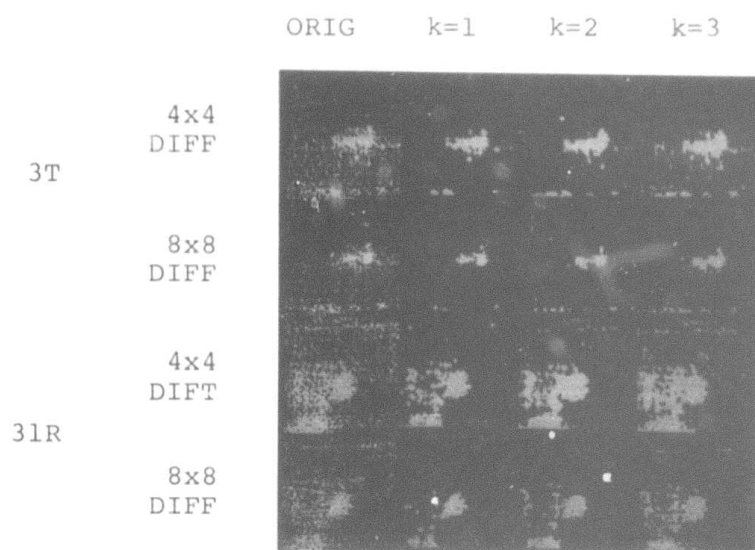
28a.
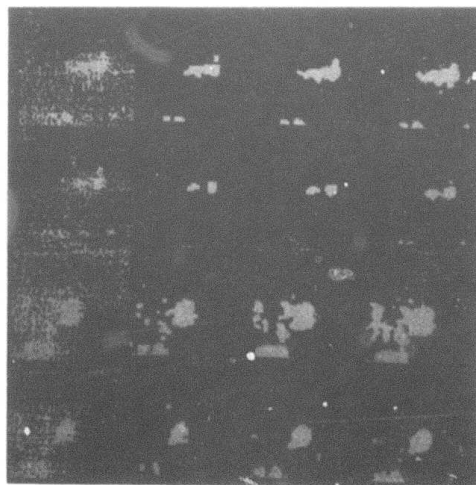
28b.

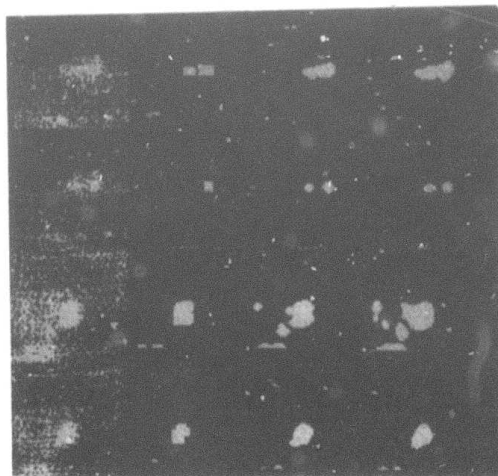Figure 28 (continued)

28c.

28d.

Figure 28 (continued)

Figure 29.   SHRINK/EXPAND of thresholded images based on
             four edge operators (ROB,3x3,4x4 DIFF,
             8x8 DIFF) and k = 1,2,3.

Key:
                    Thresholded
     Gradient   Image   S/E,k=1 S/E,k=2 S/E,k=3

     Roberts

     3x3

     4x4

     8x8
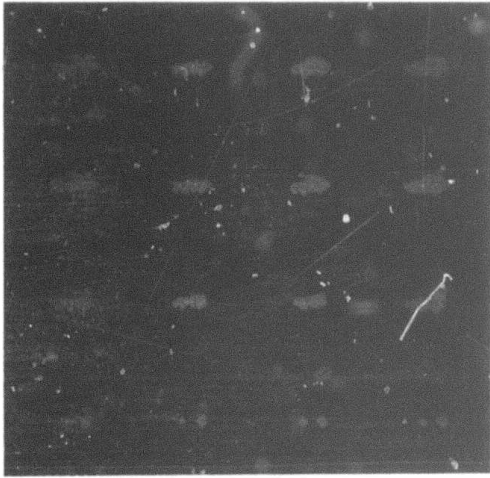
3T
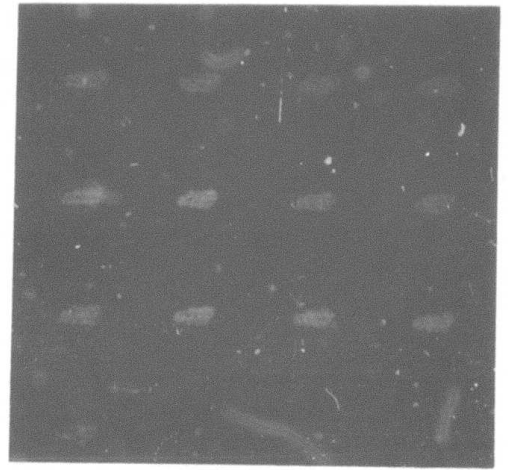
6T
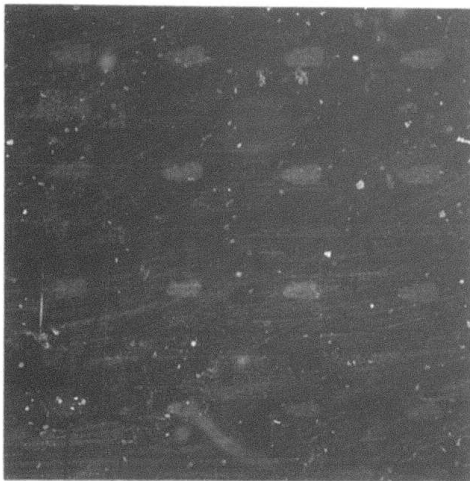
11T
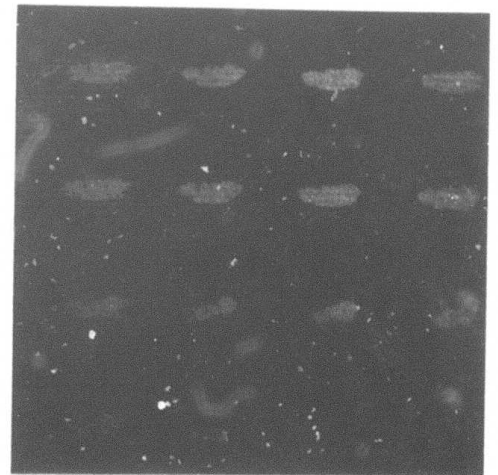
16T

Figure 29 (continued)
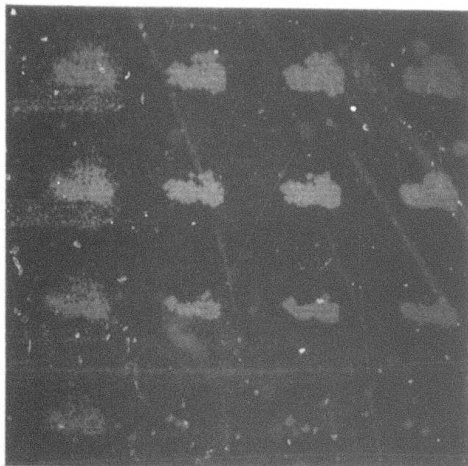
22T

28T

34T
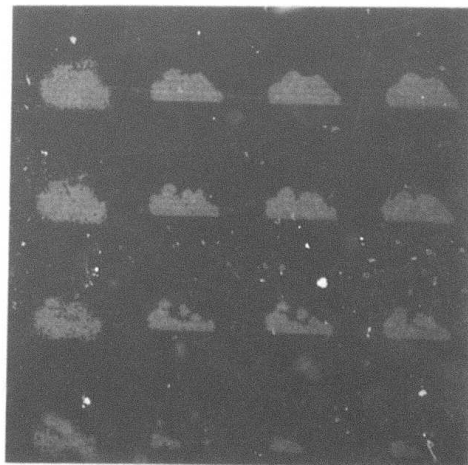
45T

Figure 29 (continued)
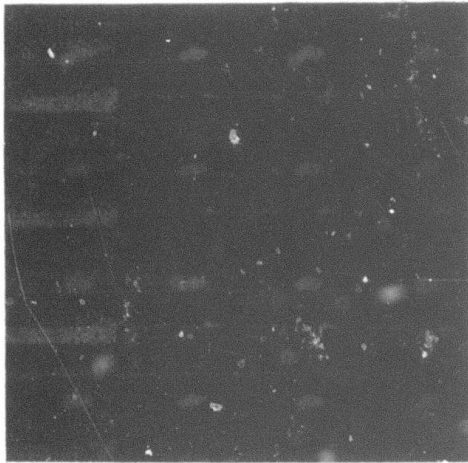
52T
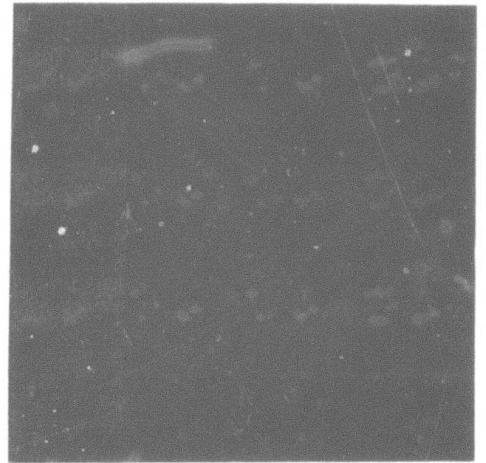
57T

3R

6R

Figure 29 (continued)

22R

24R
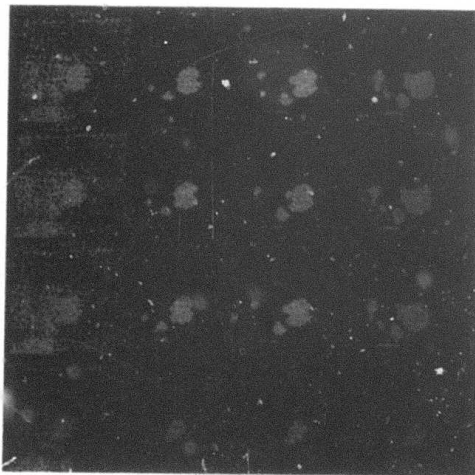
31R
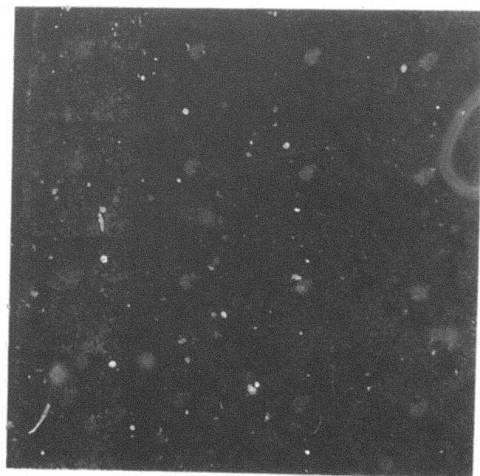
34R

Figure 29 (continued)

47R

48R

55R

57R

Figure 29 (continued)

21A

24A

27A

37A

Figure 29 (continued)

42A

44A

46A

52A

Figure 29 (continued)

54A

58A

2N

8N
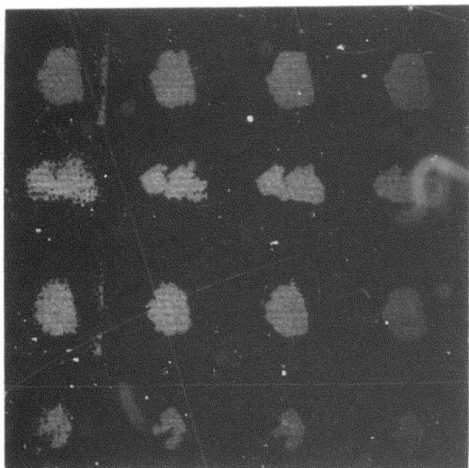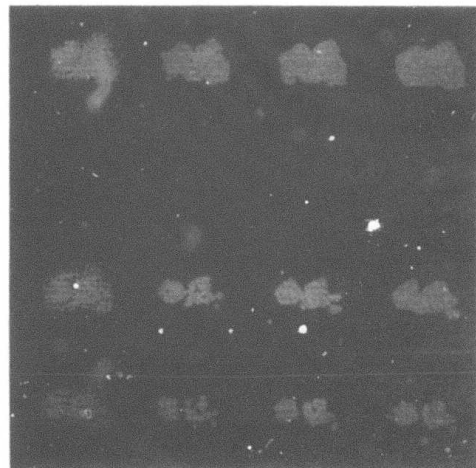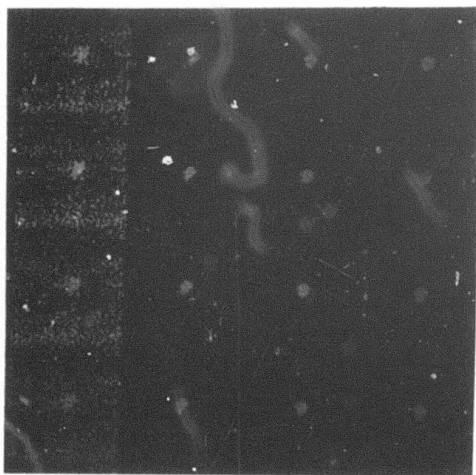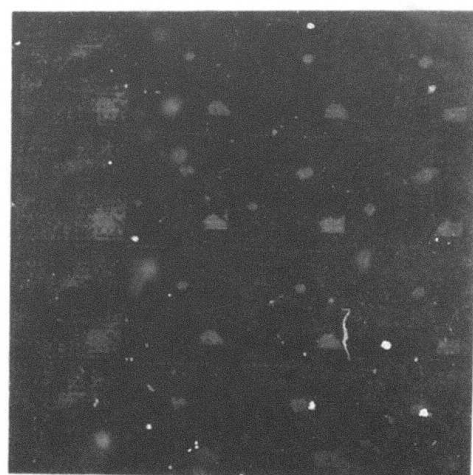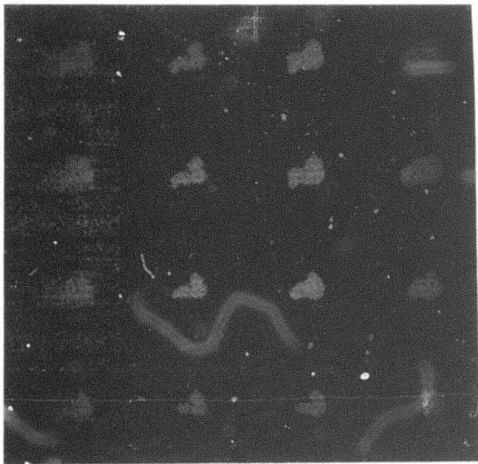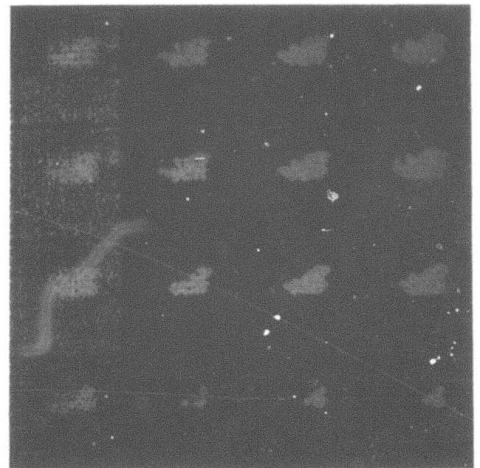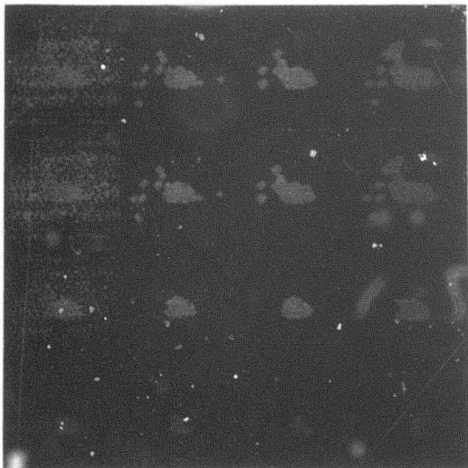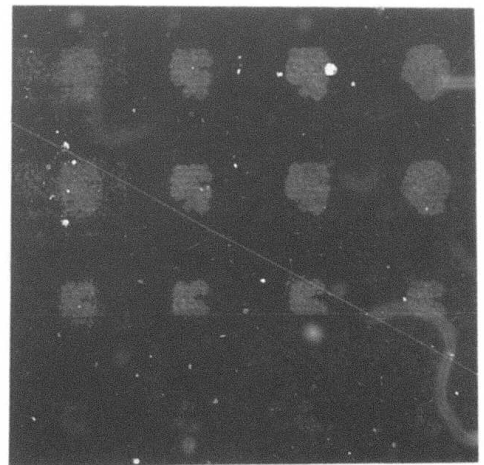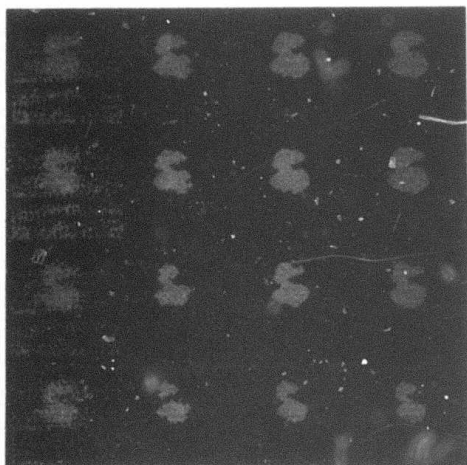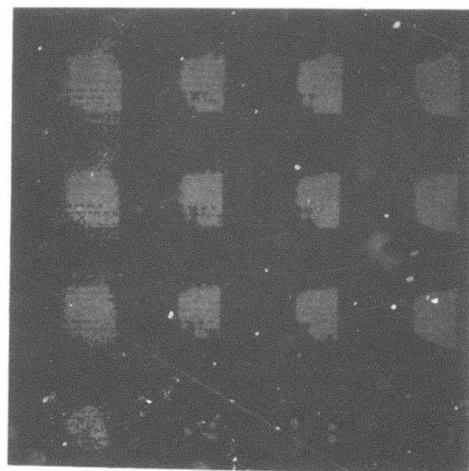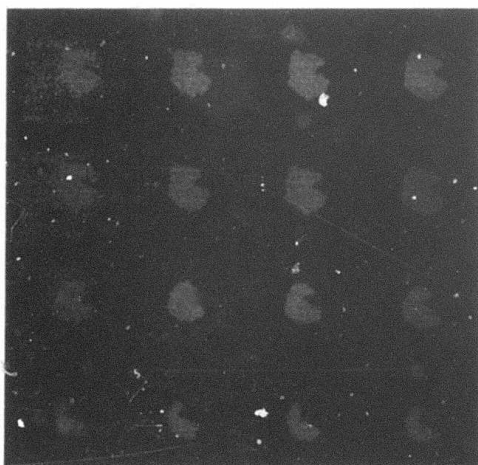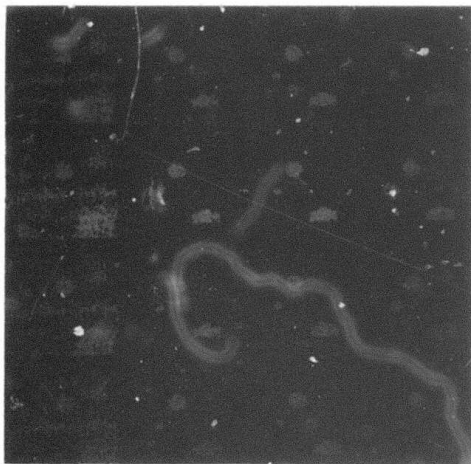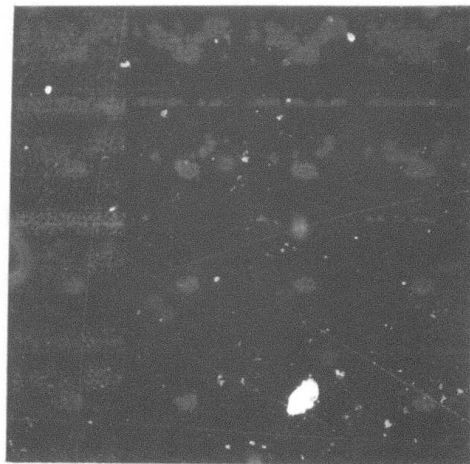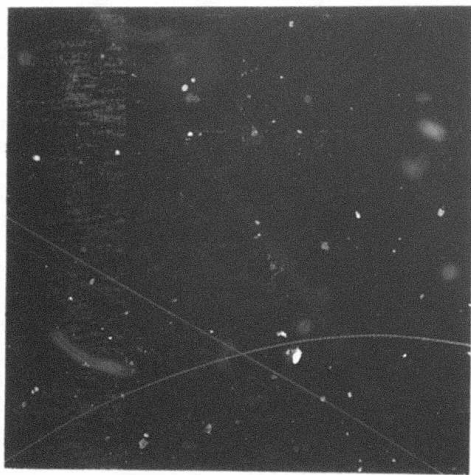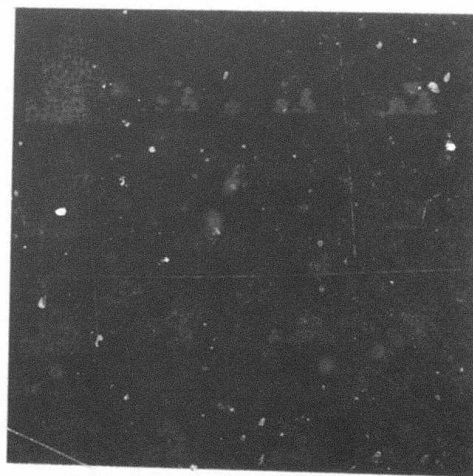
Figure 29 (continued)

14N

20N
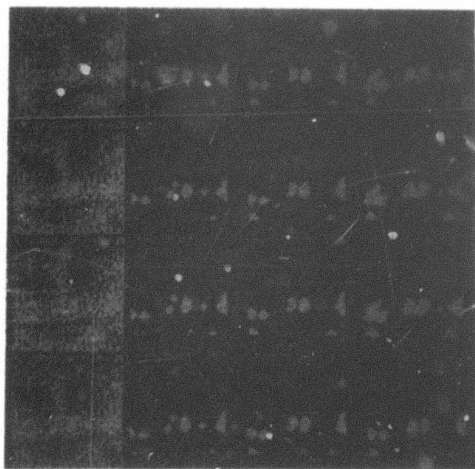
26N

32N

Figure 29 (continued)
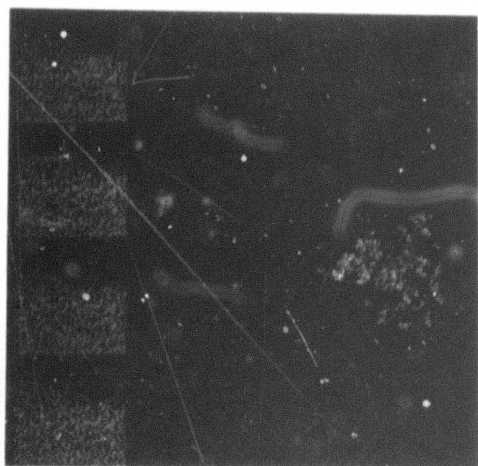
38N

44N
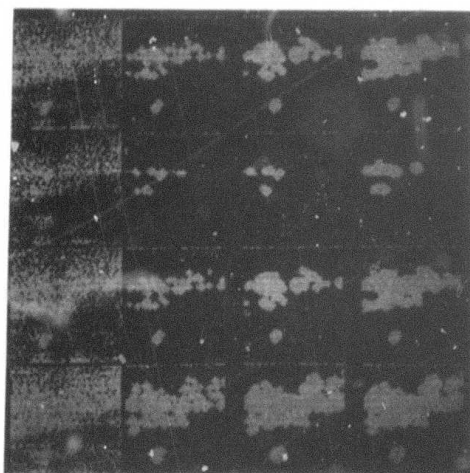
50N
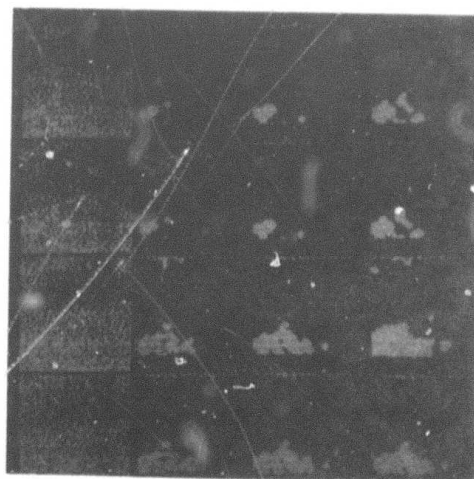
56N

Figure 29 (continued)

E4. Connected Component Analysis and Feature Extraction

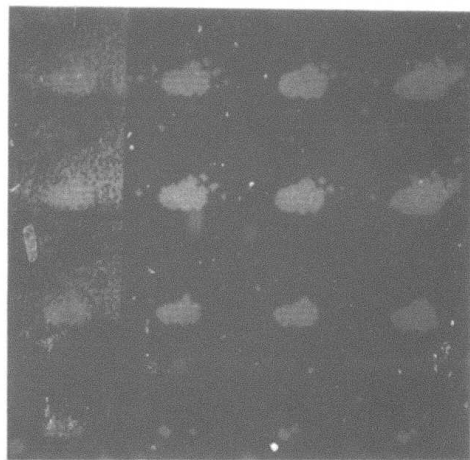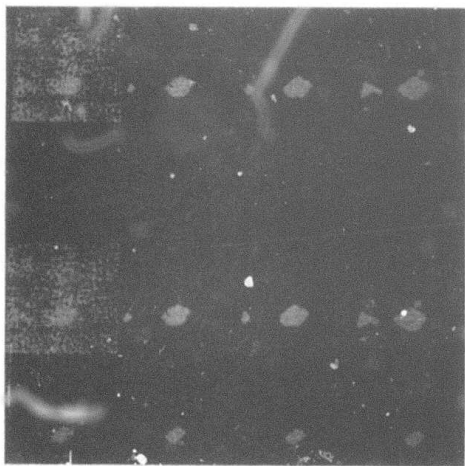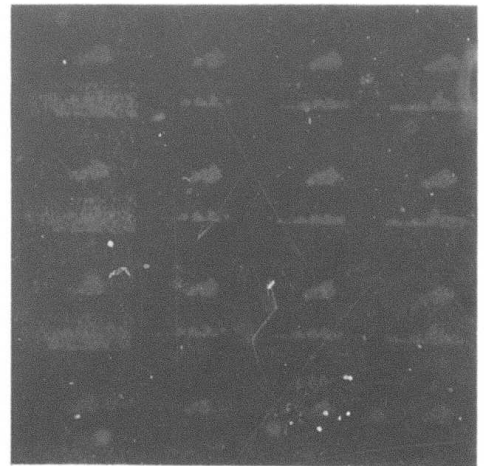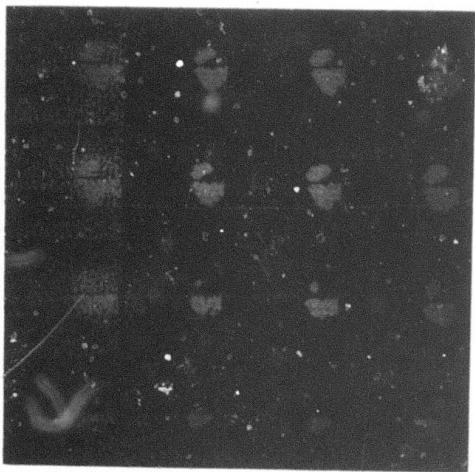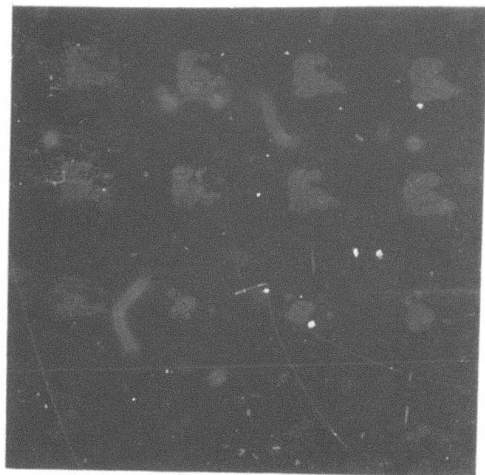The result of thresholding is a binary image. Noise cleaning operations filter this image, but it still remains to aggregate points into identified (labelled) regions. A process which labels the individual disjoint regions in the binary image, in a single raster scan, is well known in the literature [4]. It is described briefly in the following paragraphs.

A set of 1's in a binary image is connected if any two points in it can be joined by a path (sequence) of pairwise adjacent points lying in the set. A maximal connected set is called a connected component. The algorithm to be described produces the (unique) decomposition into connected components and labels the individual components (Figure 30).

As each line of the binary image is processed in turn, it is converted into a list of sequences (runs) of 1's. This list is compared term for term with the list for the previous line. Clearly any run in the current line which is adjacent to (lies underneath) a run in the previous line belongs to the same component as that previous run. Each current run which is adjacent to a previous run receives the label associated with that previous run. If it is adjacent to several previous runs with different labels then it is given one of those labels, and an entry is made in a label equivalence table indicating that these separate runs of the previous line lie in the same
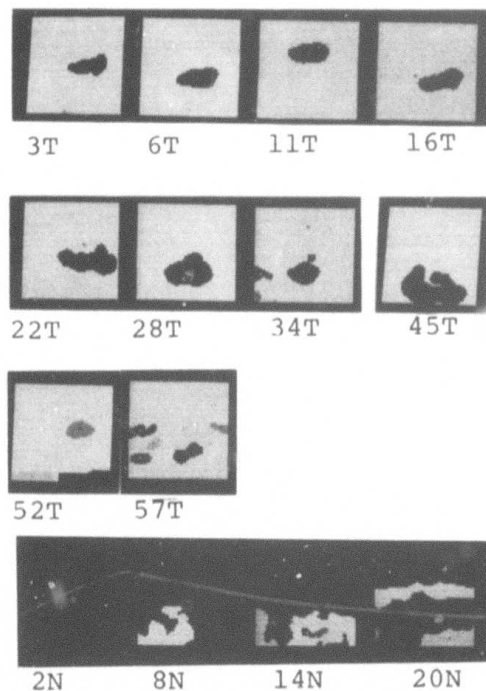
Figure 30. Components labelled with distinct solid gray levels and with centroids displayed as dots.

component. If a current row is adjacent to no previous runs, it is given a new (unused) label. When the current runs have been labeled, the scan is advanced to the next line. Once the final line has been processed, each point has been associated with a single component (possibly involving equivalent labels in the label equivalence table). A second pass can now be used to relabel each point with a unique component label, as illustrated in Figure 30.

As a binary window is being segmented, one can also process the original gray scale window (since they are in register). Various statistics based on geometry and gray level can be extracted for each component segment and accumulated during the pass. The label equivalence table can then be used to combine the statistics for component segments which belong to the same component. The set of components along with the features of each for each window were ordered by component size and stored in files for later use in classification studies. The features evaluated and stored for each component are:

1. Area (number of image points)

2. Average gray level

3. Standard deviation of gray level

4. (Average component gray level)-(Average background gray level)

F.   Discrimination and Classification

Given a set of features for objects known to lie in disjoint classes, one desires a classification rule based on the known features which will assign each object to its respective class. Various procedures are known for proposing such rules. The Fisher linear discriminant (cf. [5]) attempts to find the optimum linear projection of the feature vectors onto a line, and the optimum partition of this line, such that the ratio of between-class scatter to within-class scatter is maximized.

An implementation of the Fisher algorithm was used (as described in Section C) to discriminate noise windows from object-bearing windows. In this section, we describe experiments in classifying the components extracted in Section E4 into target/non-target classes.

Using the 13 features described in Section E4, an optimal linear classifier was trained on 30 target components and 59 noise components located in the 30 target windows. When the training set was used as a test set, two of the targets and three of the noise regions were misclassified. Attempts to omit features from the classification resulted in lower scores. For example, in trying to assess the importance of gray level and size, features 2, 3, 4, and 9 were deleted. The resultant misclassifications consisted of four targets and five noise regions (see Tables 4a, 4b).

In Section 2C, the need for an early object detection phase was stated. If non-object-bearing windows are thresholded and segmented into components, many noise regions

Feature	Fisher Direction
1	.384-01
2	-.559
3	-.285
4	-.771-01
5	-.250
6	.519
7	.287-01
8	.170
9	.471
10	.348-03
11	.103-02
12	-.355-01
13	-.116
Threshold	-.282+01

Target misclassifications: Image ref. nos. 57R, 52A

**Table 4a. Fisher linear discriminant results
on 30 targets and 59 noise regions
using features 1-13.**

Feature	Fisher Direction
1	.155-01
5	-.483
6	-.358
7	.311-01
8	-.231
10	.763
11	.135-03
12	.151-02
13	.441-01
Threshold	-.615+01

Target misclassifications: Image ref. nos. 31R,
55R, 57R, 52A

Table 4b. Fisher linear discriminant results
on 30 targets and 59 noise regions
using features 1, 5-8,10-13.

are generated which may be misclassified as targets, thus increasing the false alarm rate. An experiment verified the need for a separate detection phase. In this experiment when the noise windows were segmented and classified with the target windows (the noise windows contributing only noise components), 28 of 30 targets were recognized while 106 of 114 noise components were correctly identified. Thus of 38 objects identified as targets, 8 are false alarms (see Table 4c).

These experiments, although indicating some degree of success, were run on windows chosen by human observers as having only moderate amounts of noise. In unpreprocessed images, the likelihood of a noise window may be far greater than the likelihood of an object window, and so the object detection phase false alarm rate for noise windows (in this case, 2 out of 10) is a crucial parameter. The classification aspect of this project must be broadened in every way -- a larger data base, more informed feature selection, and a better classification strategy (e.g., Bayes) -- both in the object detection phase and in the component classification phase.

Feature	Fisher Direction
1	.305-01
2	-.580
3	-.341
4	-.141
5	-.144
6	.489
7	.148
8	-.240
9	.377
10	-.674-04
11	.979-03
12	-.105
13	-.185
Threshold	-.426+01

Target misclassifications: Image ref. nos. 57R, 52A

Table 4c. Fisher linear discriminant results
on 30 targets and 114 noise regions
using features 1-13.

3. Plans for the next quarter

A. Data Sets

The algorithms investigated up to now have been tested on a data set consisting of 40 images, selected from a larger data base of 90 frames containing 137 targets, as described in Section 2. More extensive tests, involving the entire data base, are planned. An additional data base has just been obtained, and it too will be used in future experiments. It is planned to acquire at least two further data bases which will also provide test data. The use of multiple data bases will serve as a check on the generality of both algorithms and image models. One of the data sets to be acquired will consist of real-time sequences of frames, and this will make it possible to study temporal aspects of the target detection process, including tracking of targets from frame to frame.

B.   Models

A first-approximation model for target segmenta-
tion, based on histogramming the joint occurrences of edge
values and gray levels in an image, has been developed, as
described in Section 2B. This model has suggested a number
of segmentation strategies involving classification in edge/
gray level space, rather than pure thresholding or pure
edge detection. These strategies need to be investigated.
Also, other types of models based on local property co-
occurrences should be formulated and studied. It is expec-
ted that this work will lead to an increased understanding
of the image segmentation and target detection problem.

C.   Windows

        The experiments performed during the past quarter
have employed square image windows which may or may not con-
tain targets. The distance to the ground area covered by
such a window depends on the position of the window within
the frame (as well as on the attitude and altitude of the
sensor). Normally, windows near the top of a frame will
show more distant parts of the terrain, while those near the
bottom will show closer parts, so that targets will appear
smaller near the top than near the bottom. The radiation
reaching the sensor from a window also depends on distance.
This information can and should be used in choosing para-
meter values for the algorithms that are applied to a win-
dow.

D. **Algorithms**

The algorithms described in Section 2 primarily involve edge detection and threshold selection. Other types of algorithms, e.g., for spot detection, still need to be investigated. [More advanced image segmentation techniques can also be explored, e.g., techniques based on relaxation labelling, but such techniques would be difficult to implement under the present hardware constraints.] Preprocessing techniques can also be explored, aimed at reducing the noisiness of the images to facilitate clean segmentation. Median filtering is a good example of such a technique; it is especially appropriate since histogramming is already being used in the analysis of the images.

E.   **Classifiers**

In the experiments carried out thus far, a simple Fisher linear discriminant classifier has been used. It is planned to investigate the advantages of more powerful (e.g., maximum-likelihood) classifiers. In particular, the trade-off between false alarm and false dismissal rates will be explored. Sequential decision procedures, e.g., decision trees, will also be investigated. In this connection, it is planned to make use of the Maryland Interactive Pattern Analysis and Classification System (MIPACS) in the Laboratory for Pattern Analysis at the University, which provides a wide range of tools for classifier design.

F.   Target Identification

The work done during the first quarter has dealt almost entirely with target detection and segmentation from the background. The problem of identifying targets as belonging to specific classes (e.g., tanks, trucks, etc.) must also be studied. It is planned to apply recent research results on shape description to the design of features for the target identification problem.

References

1. L. S. Davis, A. Rosenfeld, and N. Ahuja, Piecewise approximation of pictures using maximal neighborhoods, Technical Report 455, Computer Science Center, University of Maryland, College Park, MD., May 1976.

2. N. Ahuja, L. S. Davis, D. L. Milgram, and A. Rosenfeld, Piecewise approximation of pictures: further experiments, Technical Report 462, Computer Science Center, University of Maryland, College Park, MD., July 1976.

3. J. S. Weszka, R. N. Nagel, and A. Rosenfeld, A threshold selection technique, IEEE Trans. Computers C-23, 1974, 1322-1326.

4. A. Rosenfeld and J. L. Pfaltz, Sequential operations in digital picture processing, J.ACM 31, 1966, 471-494.

5. R. O. Duda and P. E. Hart, Pattern Classification and Scene Analysis, Wiley, New York, 1973, Sec. 4.10.

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	**2. GOVT ACCESSION NO.**	**3. RECIPIENT'S CATALOG NUMBER**
4. TITLE (and Subtitle) Algorithms and Hardware Technology For Image Recognition.		**5. TYPE OF REPORT & PERIOD COVERED** Quarterly rept. 1 May-31 Jul 76,
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Azriel Rosenfeld and David Milgram		**8. CONTRACT OR GRANT NUMBER(s)** DAAG53-76C-0138, DARPA Order-3206
9. PERFORMING ORGANIZATION NAME AND ADDRESS Computer Science Center Univ. of Maryland College Park, MD 20742		**10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS** 139p.
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Night Vision Lab. AMSEL-NV-VI Ft. Belvoir, VA 22060		**12. REPORT DATE** 31 Jul 76
		13. NUMBER OF PAGES 139
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		**15. SECURITY CLASS. (of this report)** Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Image understanding
Image processing
Pattern recognition
Target detection
FLIR imagery

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Techniques for detecting tactical targets on Forward-Looking Infrared (FLIR) imagery are being investigated. The principal topics covered include target and background models, object extraction and classification, and hardware technology applicable to real-time implementation.

DD FORM 1473 **1 JAN 73** EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified 403 018